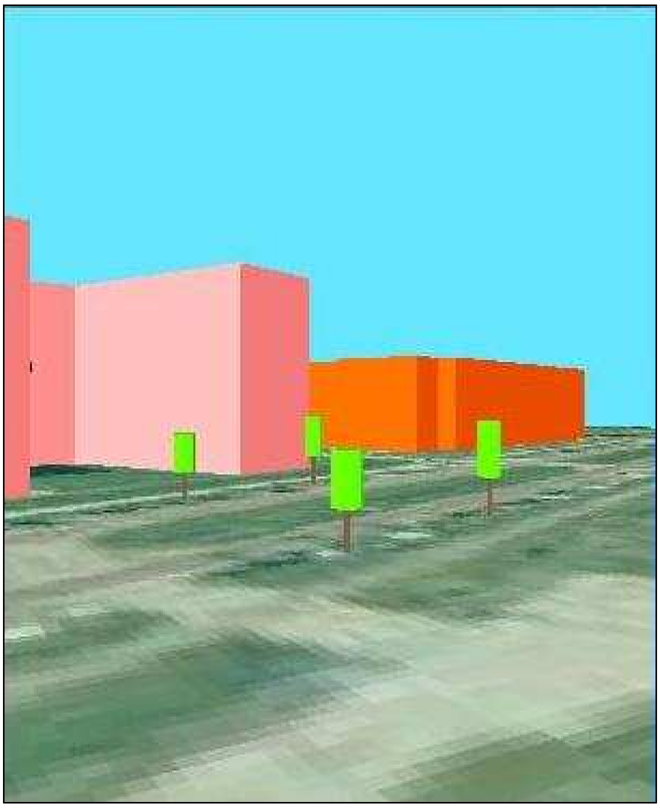
Centre for Geo-Information

Thesis Report GIRS-2006-06

Location-Based 3D Visualization Services

Zhou Han

February 2006



WAGENINGEN UNIVERSITY
WAGENINGEN UR

Location-Based 3D Visualization Services

Zhou Han

Registration number 820116-987-040

Supervisors:

Dr. Ir. R. van Lammeren
Ing. A. Bergsma

A thesis submitted in partial fulfillment of the degree of Master of Science
at Wageningen University and Research Centre,
The Netherlands

February 2006
Wageningen, The Netherlands

Thesis code number: GRS-80436
Thesis Report: GIRS-2006-06
Wageningen University and Research Centre

Laboratory of Geo-Information Science and Remote Sensing

Foreword

Time goes fast. I have been in Wageningen University for almost one and half year. By following the courses of geo-information science, I met so many classmates and worked with them. I gained a lot of group working experience with them. And the teachers are very kind and helpful. They gave me great guidance during my study here.

I want to thank the professors, teachers, lecturers and assistants, who showed me a lot in the geo-information world. Without wronging anyone, I would say special thanks to Ron and Aldo for their supervision and assistant during the past six months. They provided me ideas and helped me to revise the thesis report. Last, thanks to my parents for their support and believe.

Abstract

Rapid development of wireless communication technologies combined with powerful internet-enabled devices located by global position systems or cellular networks make the location-based services more and more popular. 3D views enable a more realistic representation of objects and service information adds value to the location-based service. In order to do this, the research defined a client-server solution for a Location-based 3D visualization application which offers 3D scenes and objects information based on the position and orientation of the handheld users.

Position and orientation extracted from the GPS data following the standard NEMA-0183 protocol were used to control the presentation of 3D scenes. 3D model was constructed mainly using extrusion of different geometric features. Objects information was provided by accessing to the server from the client devices. The testing results concluded that the position and orientation properly controlled the displaying of 3D scenes and the objects information was accessible.

Orientation should be processed more accurately and the range of the orientation degree can be adjusted by realizing the variation of the GPS data. 3D model could be automatically built and properly clipped under certain criteria.

Key words: Position, Orientation, GPS, 3D model, Objects information, Client-Server.

Content

List of Figures and Tables

1. INTRODUCTION

1.1 Background

Nowadays, wireless technology is rapidly developing which make people do not have to restrict in a fixed location to use the computing resource. And the related the commerce is becoming more and more available. The development in wireless communication is leading to provide more opportunities for commercial companies and education also. People who have the mobile devices would like to gain some information related to their locations. Wireless communication technologies combined with Internet-enabled devices constitute an ideal platform for that purpose. With the small but powerful devices which can be located by the satellite system or cellular networks [1], the Location Based Service (LBS) is feasible. Revenues from location-based service (LBS) in the European market were approximately € 108 million in 2004 and the predictions by Berg Insight indicated that LBS could contribute with over € 2 billion by 2009 in Europe [2]. In the whole world, the revenues will be around $40 billion in 2006.

Location-based services are services that are related as such or by their information contents to certain places or locations [3]. All mobile applications are based on the ability to provide remote access to data sources from mobile devices. What distinguishes Location-based Services from a pure extensibility service (such as email access from a handset) is how critical location information is to the added value to the user [4]. Services that add value by using the location component are called location-based services [5].

Mobile handheld computers and Personal Digital Assistants (PDA) such as PocketPC [6] and Palm [7] devices are now gaining acceptance. They are used in communication and provide a platform for services. They provide a way to access information that is stored locally on the PDA or accessed from a server through a wireless connection. The information can also flow in the opposite direction where the PDA is used to collect data from the construction field and send the data back to desktop applications for subsequent processing [8]. An example of commercial usage, is a tourist using a tourist information system that demands a map of the current location or descriptions of sights around him or herself [9] or other services such as city guide and navigation system. In the education usage, the PDA with a connection to GPS [10] can provide the mobile communication with teachers and other students and also location based information, specially useful for field practices, excursions etc. .

Currently, most people are using the second generation digital public mobile networks including Global System for Mobile communications (GSM), CDMA standards (Qualcomm, USA) and PDC (Personal Digital Cellular, NTT DoCoMo, Japan) [1]. For the 2G+ or 2.5G, this includes High Speed Circuit Switched Data (HSCSD), General Packet Radio Service (GPRS) and Enhanced Data Rates for Global Evolution (EDGE) [11]. All the above network generations can support LBS [12]. And the 3G networks will enable speeds up to 2 Mbps for fixed environment, 384Kbps for pedestrian and 144Kbps for vehicular traffic [13]. Universal Mobile Telecommunications System (UMTS), one kind of 3G networks, supports up to 1920 Kbps data [12].

1.2 Problem Definition

In 2004 the Vrije University Amsterdam and Wageningen University have started a joined project called MANOLO [14]. One of the main goals of the project is the wireless use of digital media, such as LBS and GPS, to support communication and community-forming in the digital learning environment. One of the methods to achieve the main goals is by the development of examples of w- and m-learning applications.

Determining the location of the user is one of the most important functions of the future mobile computing environment [9]. The technological infrastructure consists of mobile networks, mobile devices and positioning technologies. Requirements considered for the LBS domain refer to all the involved actors i.e., the mobile users, the telecom operators, the service providers, etc [1]. In this thesis work, an application is designed that is using the wireless technology to connect the client side user interface and the server side which stores the data.

In order to provide convenience to users, for instance, tourists walking or cycling through the landscapes with mobile devices can easily get relevant outlook on historic, temporary or future situations of the area. For example they can see their locations captured by GPS on the digital map which is displayed on PDA or smart phones. Nowadays, most of the landscapes and the future changes are seen in a 2D reference such as aerial photos [15]. While some implemented location-aware mobile guides use 2D maps to show the area where the user is located, pinpointing the position and providing visual information on the route the user has to follow to reach specific destinations [16]. But our real world is three dimensions (3D). A 3D display on the devices can give information [16] to the tourists that they can clearly compare the spots to what it looks like in history and what it could be in future. An example of such location system is “Roman Limes” [17]. Tourist following the limes could see what happened thousand years ago and the planned changes around in future. 3D views enable more realistic representations of changes and it is one of the added values to the users for this location-based services. Other added values could be services based on text, animation, small movies and sound. A location-based 3D visualization combined with other services using client-server mode is a solution.

Some researchers (e.g. [5, 18]) are investigating 3D graphics as a more intuitive and user friendly way to provide information on the experimental area. However, 3D visualization is not common now and people may be lack of experiences using it. Because of the small display screen and a lack of memory on the devices [8], it is still a problem how we visualize and how we provide the services. Currently, we can use devices such as PDA, mobile phones or a Tablet PC. What can be show on the screen is the landscape seen from the direction in which the user is moving. The service could be some information automatically displayed to users or by a user’s choice of interests.

1.3 Research Objectives and Research Questions

The general research objective is:

To define a client-server solution for a Location-based 3D visualization application which offers 3D scenes and objects information based on the position and orientation of the handheld users.

The general research objective is divided into research questions:

1. What components does this application consist of?
2. How to capture the position and orientation?
3. How to present the 3D model?
4. How the linked object information (texts, sound and animations, etc.) are offered?
5. How can the application, resulting from the previous research questions be tested to assess its usability?

1.4 Report Outline

In chapter 2 'CONTEXT', an explanation of location-based services is presented. Several wireless communication and positioning technologies are treated. Two 3D visualization technologies are introduced. A basic description of Client-Sever mode is finally given which provide the objects information related to the location of the users' handheld (Question 4).

In chapter 3 'CONCEPT', two important questions are answered. One is how to get the proper position and orientation of the user's handheld (Question 2). The second one is about presenting the 3D model (Question 3).

In chapter 4 'DESIGN', the application architecture is designed. Next, three kinds of handheld devices are compared. Then how the 3D data and objects information data are stored and processed are explained. The design of user interface and use of wireless network is introduced before the list of application approach steps.

In chapter 5 'IMPLEMENTATION', it starts from the choices of hardware and software, and then tells about the decision based on the concepts and design. Finally how each it is implemented in the application.

The last two chapters are TESTING, and CONCLUSION , DISCUSSION and RECOMMENDATION discuss subsequently the research questions 5 and the original research objective for the whole application. The testing focuses on the usability of the application.

2. CONTEXT

In order to answer the research questions for a location-based 3D virtual outdoor world, the definition of LBS should be clear. So a definition is given and three generations of LBS are listed together with their related services. The figure 2.1 gives an overview of LBS and also shows the main components included in a location-based service. It involves wireless connections for accessing the data, so several mobile technologies from 2G via 2.5G until 3G are described. It also relies on geographic information and position technique. Positioning techniques are treated and 3D visualization is mainly explained.

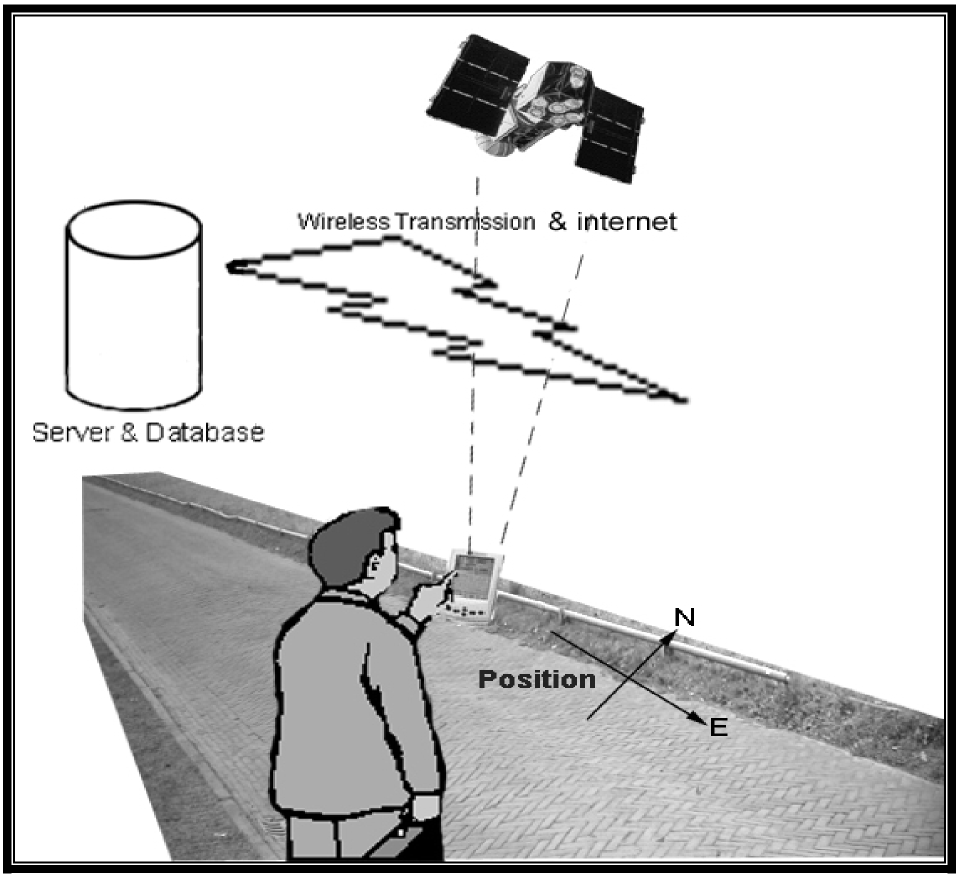


Figure 2.1 Overview of Location-based service

A description of client-sever mode is given because the data are partially stored at server side, the client-sever can use request and response for data transmission. The client side in this mode is called thin client, whereas processing and storage capacity take place at the server, meaning the client doesn't need to "carry this capacity" by his own device.

2.1 Location-based Services

The integrated technology of geographic information systems (GIS), Internet, wireless communications, location determination, and mobile devices has given rise to exciting new types of information utilities that may be referred to simply as location-based services. These systems are making a major impact on how we navigate our world and how business is done. Knowing where a person or object is at any time offers a powerful and new kind of information. Location services deliver geographic information between mobile and/or static users via the Internet and/or wireless network [19]. Location-based service is very useful because most of our datasets are linked to location or geographic components. So it supporting communications and computing technologies can provide access to this information in wide range of applications for business, consumers and government market sectors.

A location-based service, in the broadest sense, is any service or application that extends spatial information processing, or GIS capabilities, to end users via the Internet and/or wireless network [19]. There are also other definitions to location-based service, the Pull, Push and Telemetry. The table 2.1 gives an overview of them.

There are three generations location-based services identified by Gravitate, Inc. (2000).
First generation services require the user to manually input location in the form of a street address or postal code and are typically available to stationary desktop computers or mobile units. Second generation location services, which are available today, have the ability to automatically determine rough locations on the postal code level. Third generation location services are more location aware, taking advantage of more precise positional information and have the capability to initiate services proactively based on location. It can notify users of relevant events or conditions without active participation of the user.

Type of Definition	Type of service
PULL — Information flow that is expected by user [20]. Services that use the geographic position of a wireless device to derive information related to that location enable users to "pull" information to them wherever and whenever it is needed. The user is the (active) actor.	▪ Travel Directions — *I am here, how do I get there?* ▪ Taxi Hailing — *I need a taxi now* ▪ Mobile Yellow Pages — *Where is the nearest x?* ▪ Buying Services — *Notify me when I am near a supplier that carries the specific item I have been looking for* ▪ Instant Information
PUSH — Information flow that is not expected by the user [20]. Use the location to qualify the holder as a potential customer or recipient of a service. The user is the passive receiver.	▪ Mobile Advertisements — *Privacy issue aside* ▪ Friend Finders—*SMS(short message service)* ▪ Zone Alerts — *Track movement* ▪ Traffic Alerts — *Status of predefined travel routes*
Telemetry — Machine-to-machine communication enables distributed assets to automatically notify service providers of their location and status.	▪ Always applied to fixed assets such air conditioner, copy machine

Table 2.1 Three types of definition

2.2 Mobile Communications

Location-based services require a stable and fast connection between the service providers and the devices. The data transmission can include maps, pictures, text, and video stream, etc. Some information can be preinstalled in the client devices but most of time, real-time information needed when those data are stored remotely, like in a central server.

GSM, globe system for mobile, is the most widely used mobile network all over the world. This cellular network is the second generation network and is mainly used for phone with voice communications. By the late 1990s, wireless industry undertook the task of defining new wireless systems-3G, was to be based on packets of data. Three new wireless standards emerged: CDMA2000 (evolution of IS-95), EDGE (evolution of GSM for existing spectrum) and WCDMA (evolution of GSM for new spectrum using a 5-MHz WCDMA carrier).

The evolution of GSM to 3G is about gradually adding more functionality to the existing GSM network and business. It begins with an upgrade of GSM network to 2.5G by introducing GPRS technology and EDGE is a further developing step of GSM packet data. EDGE can handle about three times more data subscribers than GPRS. With the continuation of EDGE standardization towards GERAN (GSM/EDGE Radio access network), a full alignment of WCDMA will be achieve. The emerges of these evolutions from GSM to GPRS, EDGE and WCDMA is a seamless 3G UMTS (Universal Mobile Telecommunications System) Multi-Radio network, which maximizes the investments in GSM and GPRS. A wireless world forum predicted in 2005 that 3G revenues will increase to €157 billion by 2009. Predictions may vary but the consensus is clear that the result will be positive for 3G services [21].

UMTS offered data rate targets are: 144 Kbps satellite and rural outdoor, 384 Kbps urban outdoor and 2048 Kbps indoor and low range outdoor. Its network services have different quality of service classes for four types of traffic: Conversational class (voice, video telephony, video gaming), Streaming class (multimedia, video on demand, web cast), Interactive class (web browsing, network gaming, database access), Background class (email, SMS, downloading) [22].

The 4G technology has already being on the way. It will integrate radio and television transmissions, and consolidate world's phone standards into one high-speed technology [23]. The 4G network has the ability to roam across different wireless network standards with the one device and has a higher level of bandwidth, a reasonable figure to expect is about 20Mbps.

2.3 Positioning Techniques

There are several options available with current positioning techniques for delivering LBS. These technologies and techniques are listed in table 2.2. The main element in the network design is the degree of accuracy when point a user's location.

The telecom based technology cell ID is not accurate enough [24]. The diameter of a cell ranges from several hundreds of meters in urban areas to three kilometers in agricultural areas. Sometimes a cell can be divided into sections enabling a reduced cell size of up to two thirds. But it is rather cheap and can be used both out- and indoor. Considering the other three, GPS requires either a GPS receiver connected to mobile device, or a device with a built in GPS receiver. Besides the higher investment costs for equipment the GPS technology delivers a higher accuracy. Considered the location-based application developed in this research requires high accuracy, a GPS receiver will be used to determine the Tablet PC or PDA's position. Nowadays, a GPS receiver always integrated an internal compass which can be used to display the direction of user and indicate the direction of next steps. The position and orientation data which are the most important factor for navigation in a 3D world can be gathered.

Type	Methodology	Pros	Cons	Industry Applications
Cell Identifier (cell ID)	Base station uses radio frequency signals to track mobile device	Relatively widespread infrastructure	Hard to pin down user's exact location to a few meters	Wireless network providers, police force, banking government security, welfare
Global Positioning Systems (GPS)	24-satellite network	Outdoor precision within five-meter range	— Expensive — User device must be in direct line of sight — Device needs special embedded chips	Military applications, commercial applications like real estate, security, police force (not as successful in consumer settings)
Assisted Global Positioning Systems (aGPS)	Enhancement over GPS Perpetually locates device and coordinates data flow, unlike GPS	— No "cold starts" — Faster fix on location	The same as above	The same as above
Broadband Satellite Network	Relies on low earth-orbit satellite architectures to create a global network	Lower signal latency with user devices	Complex to maintain	The same as above

Table 2.2 Mobile Network Platform [24]

More accurate solutions for location based on wireless local area networks is suitable, but these are only suitable for rather small area. Radio Frequency Identification both active and passive (RFID) are proper for accurate position within buildings. Ultra Wide Band (UWB) seems to be the right technology for all the service but unfortunately it is not available yet [25].

2.4 Three-dimensional Visualization

2.4.1 VRML

Three dimensions (3D) visualize the geometric shapes like it appears in the real world. It gives us a better comprehension of the objects compared with the 2D visualization, for instance, in a lot of PC games, 3D is the main displaying way for players. For location- based 3D visualization, a format used for displaying 3D model on desktop computers is the Virtual Reality Modeling Language (VRML) [26], which is a scene description language for representing 3D interactive model on the web. VRML models are displayed in VRML plug-ins [27] for Netscape or Internet Explorer. A VRML browser can also be developed for PocketPC, but the problems of running 3D in PocketPC are its small screen, low memory and not very powerful processor. In the ISO standard [28, 29], it defines the base functionality and text encoding for VRML, and the base functionality and all bindings for the VRML External Authoring Interface.

VRML is a file format for describing interactive 3D objects and worlds. VRML is designed to be used on the Internet, intranets, and local client systems. It is also intended to be a universal interchange format for integrated 3D graphics and multimedia. VRML may be used in a variety of application areas such as engineering and scientific visualization, multimedia presentations, entertainment and educational titles, web pages, and shared virtual worlds.

The VRML models can be exported by CAD programs using ‘IndexFaceSets’ and geometric transforms to represent all of the geometry in the model. It is accurate visual representation but its file size is large and processing of model is not efficient. ArcScene can also be exported as VRML files. The VRML file will look exactly like the 3D scene, including all features and symbols, offsets, and extrusions [30]. The VRML scene's symbology cannot be altered in the web browser. VRML files can take a long time to generate, and the files can be very large. TINs are generally far more compact than grids when exporting to VRML.

2.4.2 X3D

X3D is the successor of VRML. It is an Open Standards XML-enabled 3D file format to enable real-time communication of 3D data across all applications and network applications. It improves upon VRML with new features, advanced API’s, additional data encoding formats, stricter conformance, and a componentized architecture using profiles that allows for a modular approach to supporting the standard and permits backward compatibility with legacy VRML data. It can be used across hardware devices and in a broad range of applications including CAD, visual simulation, medical visualization, GIS, entertainment, educational, and multimedia presentations.

X3D provides both the XML-encoding and the Scene Authoring Interface (SAI) to enable both web and non-web applications to incorporate real-time 3D data, presentations and controls into non-3D content.

In ISO standard [31], it specifies the encoding of the functionality using the technique used by VRML. The "Classic VRML" encoding which can play most non-scripted VRML (version: 2) worlds with only minor changes. None of the technology has been lost; instead, it has evolved into X3D. X3D has made a very large effort to maintain as much compatibility with VRML as possible. There are some X3D viewers, browsers and plug-ins, commercial and freeware. The Cortona VRML Client 5.0 has the expanded range of supporting X3D nodes.

2.5 Client-Server

Client-server describes the relationship between two computer programs in which the client makes a service request from another program, the server, which fulfills the request. Although the client-server idea can be used by programs within a single computer, it is a more important idea in a network. In a network, the client-server model provides a convenient way to interconnect programs that are distributed efficiently across different locations. For example, to check the train schedules from your computer, a client program installed at the client computer forwards the request to a server program at the a central train schedule server, that responds the timetable back to the client.

In the usual client-server model, server is activated and awaits client requests. Typically, multiple client programs share the services of a common server program, which means multiple accesses.

The client programs and server programs are often part of a larger program or application. Considering this situation, an application uses the web browser or application based browser to requests services (the sending of Web pages or information from center database) from a Web server (Hypertext Transport Protocol or HTTP server) in another computer somewhere on the Internet or just in the local area network. The request can include some parameters, like object IDs, etc, then sever will do some queries or calculations on a database or a geo-database according to those parameters to fulfill the request. Finally, the server will send the results back to client. In the Location-based services, this kind of mode is also called as thin-client mode.

3. CONCEPT

Position can match the word "location" in location-based services. However, within a 3D visualization, only position information is not enough for an application to decide which landscape scene should be displayed. As a result, the combination of position and orientation is needed to provide necessary "location" information. On the other hand, 3D model is also important which can match the word "visualization". The integration of position, orientation and 3D model make the main function of location-based 3D visualization feasible. The figure 3.1 shows my concept for the location-based services. The numbers 1 and 2 in this figure are what this chapter will introduce.

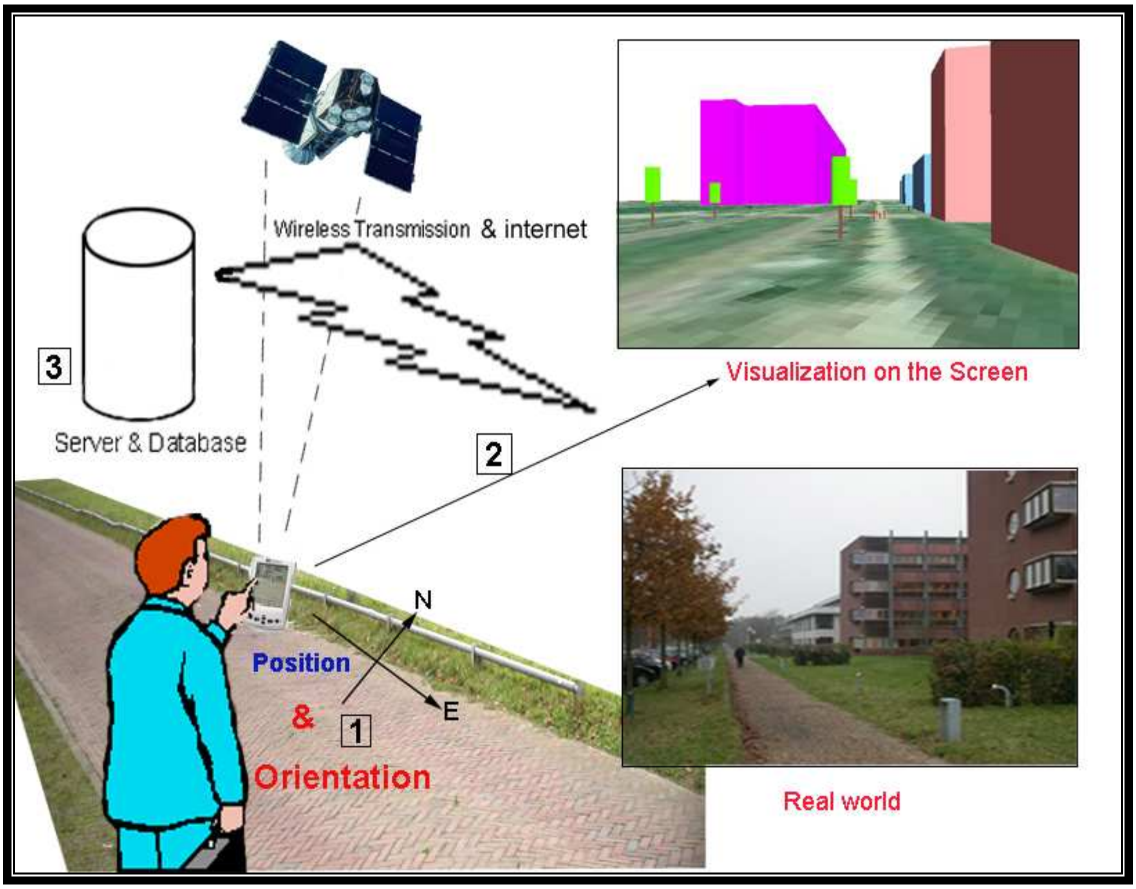


Figure 3.1 Concepts of Location-based 3D visualization Service

3.1 Position and Orientation

When there are only 2D maps to be displayed on the device, only position data is enough to locate the places of landscape or persons. In some navigation systems, when you move, the 2D maps will always keep your heading in front you which means the maps are turning according to your movement. What you see on the screen is that you are always heading forwards.

When coming to the 3D scenes, most of the time, people would like to see the landscape in front of them or on their left or right. How to get the scenes in front of or beside the users when their positions are known? To simplify this question it looks like how to 'turn left or right' in the 3D scenes. The orientation is the answer which can indicate the left or right movement. So when position and orientation are obtained, with the predefined eye's level, our 3D visualization comes true.

3.1.1 National Marine Electronics Association (NMEA 0183)

Position and orientation are both key parameters which correspond to the word "location" in 3D. A GPS receiver with internal compass should be used to get both location and orientation and the NMEA interface is the protocol used to communicate between the GPS and the application. The NMEA 0183 is most common used standard.

The NMEA 0183 Interface Standard defines electrical signal requirements, data transmission protocol and time, and specific sentence formats for a 4800-baud, 8 data bits, no parity and one stop bit data bus [32, 33]. There are different NMEA format, such as NMEA 0180, NMEA 0182 but NMEA 0183 is most often used.

NMEA 0183 sentences are all in ASCII. Each sentence begins with a dollar sign ($) and ends with a carriage return linefeed (<CR><LF>). Data is comma delimited. All commas must be included as they act as markers even if that field is not available. Some GPS do not send some of the fields. A checksum is optionally added (in a few cases it is mandatory). Following the $ is the address field AACCC. AA is the device id. For instance, GP means globe positioning system receiver, LC means Loran-C receiver. CCC is the sentence formatter, otherwise known as the sentence name. For instance, RMC means Recommended Minimum Specific GPS Data that can provide position, orientation and speed while GGA indicates the globe positioning system fix data.

3.1.2 GPS Data

In a location-based service, we are mainly using the fixed GPS data and also the data which calculates the traveling direction. In this section, we will first introduce the format of two main data types used.

GGA data:

*$GPGGA, hhmmss.ss, llll.ll, a, yyyyy.yy, a, x, xx, x.x, x.x, M, x.x, M, x.x, xxxx*hh*

RMC data:

*$GPRMC, hhmmss.ss, A, llll.ll, a, yyyyy.yy, a, x.x, x.x, ddmmyy, x.x, a*hh*

Data is comma delimited. All commas must be included as they act as markers even the data in that field is not available. Table 3.1 and 3.2 shows the meaning of number in each type with examples:

No.	$GPGGA	Example *$GPGGA,143915,5159.2804,N,00539.9544,E,1,06,01.3,13.8,M,43.7,M, ,*5E*
1	UTC of Position	*143915*
2	Latitude	*5159.2804*
3	N or S	*N*
4	Longitude	*00539.9544*
5	E or W	*E*
6	GPS quality indicator (0=invalid; 1=GPS fix; 2=Diff. GPS fix)	*1*
7	Number of satellites in use [not those in view]	*06*
8	Horizontal dilution of position	*01.3*
9	Antenna altitude above/below mean sea level (geoid)	*13.8*
10	Meters (Antenna height unit)	*M*
11	Geoidal separation (Diff. between WGS-84 earth ellipsoid and mean sea level. -=geoid is below WGS-84 ellipsoid)	*43.7*
12	Meters (Units of geoidal separation)	*M*
13	Age in seconds since last update from diff. reference station	
14	Diff. reference station ID#	
15	Checksum	**5E*

Table 3.1 GPS data format GGA

So, $GPGGA,143915,5159.2804,N,00539.9544,E,1,06,01.3,13.8,M,43.7,M,,*5E, means the data was taken at 14:39:15 UTC with fixed latitude 51^o and 59.2804’ North and longitude 5^o and 39.9544’ East. Six satellites were available and the altitude is 13.8 meters.

No.	$GPRMC	Example *$GPRMC,143920,A,5159.2805,N,00539.9547,E,1.07,149.95,091105,,,A*68*
1	UTC of position fix	*143920*
2	Data status (V= warning, A=ok)	*A*
3	Latitude of fix	*5159.2805*
4	N or S	*N*
5	Longitude of fix	*00539.9547*
6	E or W	*E*
7	Speed over ground in knots	*1.07*
8	Track made good in degrees True	
9	UT date	*091105*
10	Magnetic variation degrees (Easterly var. subtracts from true course)	*149.95*
11	E or W	
12	Checksum	*A*68*

Table 3.2 GPS data format RMC

*$GPRMC,143920,A,5159.2805,N,00539.9547,E,1.07,149.95,091105,,,A*68* means the data is valid and was take at 14:39:20 on 9th of November, 2005 with latitude 51^o and 59.2805’ North and longitude 5^o and 39.9547’ East. The orientation is 149.95^o (360^o means straight north, 180^o is straight south).

With this GPS data ($GPRMC), after extracting the useful data, position and orientation problem is solved. Orientation can only get when moving, when static, the orientation stays the same as before the stop.

3.2 Construction of 3D Model

There are some articles that introduce how to construct cities models automatically [34]. It involves the integration of heterogeneous data source which allows the access in a common way, and then interpreting and extruding the 2D data into three dimensions. The location and geometric properties of the objects should also be properly used to match the objects in different data source.

An easy way to generate the 3D geo-data is the extrusion from the objects footprints into simple block models. The most important thing is to get the height information. The height value can be depended on the number of floors while the lower edge can be derived from the digital elevation model (DEM). The height information can be found either in a separate file containing the DEM data or from the one field in the attribute table. After extrusion, a point will be turned into a vertical line, lines into walls with the vertical orientation and polygons into blocks. A natural filtering criterion can be imported based on proximity [16] to simplify the generation. Not all the buildings, trees, road lights, labels will be visualized in three dimensions. For a better execution, features may not completely be the same as the one in the same world. The shape and location of the objects are the same, but the textures may not truly representative. This is a selection during the construction and is based on the accuracy we want and the actual transmission speed (more accurate, larger size).

Another way of generation is to use CAD programs with photogrammetry to construct the 3D model and then export them to VRML [35]. But only this is not enough, there are errors on some objects surface from perspective view. Simplification algorithms, feature detection and removal are involved in order to construct the 3D model [34]. The VRML city model downloads in moderate time over UMTS or ISDN or faster lines. It renders smoothly with modern PCs and can be further improved by defining several level-of details (LOD). The download and rendering speeds could also be improved by providing a low-detailed model with coarser or no textures. Users with poor 3D graphics capabilities or modems are provided with web pages containing only 2D maps.

The geo-reference used is the same as the original data we used (e.g. “RijksDriehoekstelsel”---Dutch grid). The entire 3D model is constructed using this coordinate system. When converting the objects to VRML, the original coordinate system is transferred to a “VRML” own coordinate system. The original geo-reference will be lost. VRML file has its own specification. The 3D model still has its unique positions but with different coordinates. The mathematic relation between the original grid and the VRML grid will be introduced in the next part.

3.3 Clipping of 3D Model

When users are using a location-based 3D visualization, they are usually interested in the landscape around them, such as the nearby buildings or forests, rivers, etc. So a clipping of 3D model is necessary because only a part of the whole model has to be displayed in order to match the real world. In the client-server mode, the server will load the whole 3D model but only send the clipped part to the client on basis of position and orientation. This can also save the responsible time caused by program calculation and internet or wireless transmission of data.

The clipping is happened during the run time of the application. The whole 3D model is loaded but only the clipped part of the model is rendered. When the valid GPS data are received by GPS receiver, a 3D responsible area can show up in the screen. The definition of this 3D area is determined by the position and orientation extracted from the GPS data. The user himself or herself is considered as one facet of the area. The figure 3.2 below shows how it is defined.

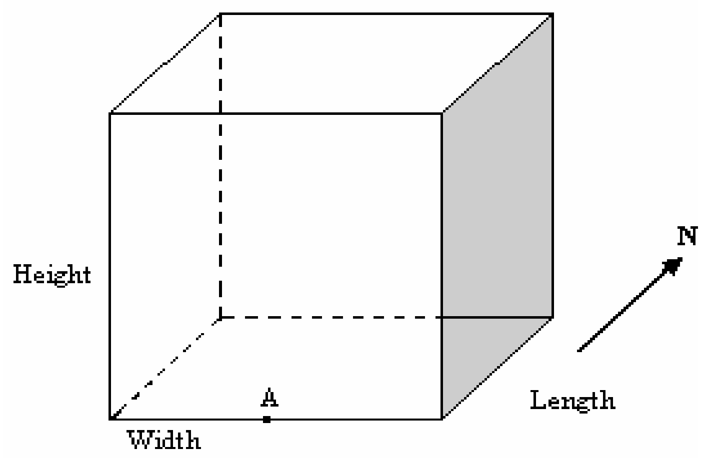


Figure 3.2 Definition of Cubic

Suppose the user is standing in Point A which is in the center of that line and her orientation is facing to the North, what the user will see is an area (= width*Length map units^2) in front of her eyes with a fixed height. The exact number of the width and length can be changed depending on the size of the real world in the dataset. Normally, they are measured in meters. And the height of the user, which is also said as eye-level, in point A can be changed from the foot (1m) to eye's height (average 1.75m) or even much higher. The change of the height will give different views of 3D scenes to users.

In the VRML file, the original X axis matches the x axis in VRML, and original Y axis matches the negative z axis. And the y axis in VRML is used to define the height (see table 3.3).

Geo-reference	**VRML**
X axis	x axis
Y axis	-z axis
Z axis	y axis

Table 3.3 Axis relationship

So the X, Y coordinates in the geo-reference coordinating system can't directly be used in the VRML file. A conversion was applied by using the original X, Y to subtract the X, Y coordinates of the center point. For example, a point in the Dutch grid coordinate system has the value (174198.563, 444478.757) and the center in the research area is (174099.000, 444298.000). After a conversion, a VRML coordinate should be (99.563, y, -20.757) considering the axis relationship. But the computer still doesn't know which part of the 3D model should be displayed. Because only getting position is not enough to tell the computer to clip the suitable world at certain points. Each point has 360^o views so orientation is used to determine which part will display. As it was said in chapter 3.1.2, orientation is a degree number between 0^o~360^o. In this concept, the 360^o is divided into 12 zones each of which includes 30^o. Each zone indicates one direction so there will be 12 different directions in total. If the received degree of orientation is within the same zone, the displayed 3D scenes are the same. More zones can be defined that make the 3D scenes better match the real world, like 24 zones(15^o/zone, 24 different directions), 36 zones(10^o/zone, 36 directions) or even 360 zones (1^o/zone, 360 directions) according to the complexity of the 3D model.

After the introduction of the concepts and considering the technology in chapter 2, now figure 3.1 can be discussed in detail. The data from the GPS receiver will be used to extract the position and orientation by using the NMEA-0183 protocol, which is widely used in the GPS transmission. The transmission between GPS and client devices is done by Bluetooth technology. The 3D model will be constructed in VRML format with its own geo-reference system and the model is stored on the server. With the position and orientation, the clipped 3D model will be sent to client and displayed on the screen of the client devices. The UMTS or GPRS technique, each of which has high transmission speed, is used for communication by sending the request and responses between client and server. The client-server mode is quite suitable if there are a lot of users accessing the same 3D model.

4. DESIGN

This chapter will describe what the application consists of. Due to the specific purpose---client-sever mode of location-based 3D visualization. The requirement of this application includes the hardware with 3D supported browser, 3D model, and software to develop the user interface and connect the client-server via wireless. Figure 4.1 shows the application structure and the numbers 1, 2 and 3 also match the numbers in figure 3.1 which had been introduced before, those are position, orientation, 3D model and client-server request and response. Finally, several phases are listed in the final part which shows the application approaching steps.

4.1 Application Architecture

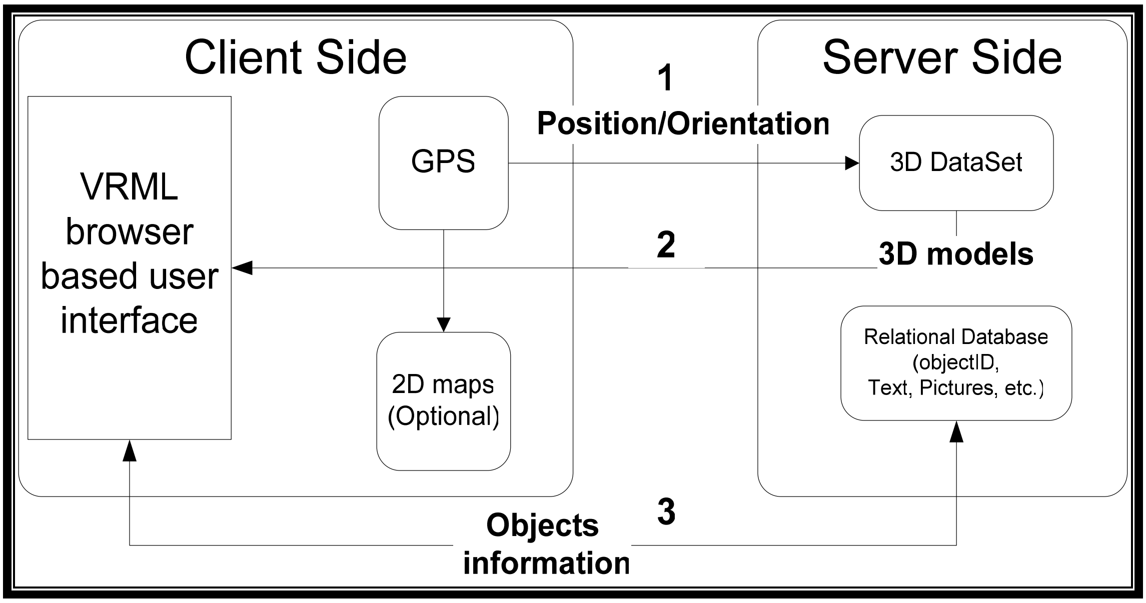


Figure 4.1 System Architecture

There are some researchers exploring a prototype system of location awareness visualization of a 3D world. Some researchers (e.g. [5, 18]) are also investigating 3D graphics as a more intuitive and user friendly way to provide information on the experimental area. However, limitations of current mobile devices make it difficult to obtain a smooth navigation of 3D representations, especially when complex environments (e.g., full cities) are taken into consideration.

The application consists of several components, including user interface, GPS component, 3D data and information for some objects. The users will see the user interface after clicking the icon of the application as installed on a handheld. And it will display the 3D scenes and offer the required objects information. From the user interface, users need to activate the GPS. They can also change the height of eye-level if they want. The description for these components is listed below:

Components	Function description
User interface	● Activate the GPS to let it work; ● Set the height of eye-level; ● Displaying position and orientation related 3D landscape scenes on the screen; ● Displaying the user interested information predefined by the developer requested by users.
GPS	● Obtaining GPS data; ● Providing other components with the actual position and orientation.
Objects Information	● The URL of the information is stored in a database on a server; ● Getting data from the right place by reading the URL.
3D data	● 3D data are made from ArcScene or 3D Max.

Table 4.1 System components and description

4.2 Devices

Most of the new mobile phones and PDA have already support for GPRS, Bluetooth and MMS, building camera and GPS components, as well as sophisticated applications, while Tablet PC is an extended desktop computer. The description of devices is listed in table 4.2.

In the application, Tablet PC is firstly used because it doesn't have hardware limitations and it also has wireless connections whereas PDA and smart phones have limited screen size, processors, memory, and 3D display capability. Then we propose to move the application to the PDA with necessary changes. The GPS can connect to both devices via Bluetooth. The table 4.3 gives the options for the server and client devices.

Devices	Description
PDA	Personal Digital Assistant: A portable computing device for organizing personal data such as telephone numbers, appointments, and notes. The PDA we use can connect the GPS device by Bluetooth and use Windows CE operation system.
Tablet PC	A tablet PC is a mobile computer shaped in the form of a notebook or a slate with the capabilities of being written on through the use of digitizing tablet technology or a touch screen. A user can use a stylus or pen-enabled pen to operate the computer without having to have a keyboard or mouse [12]. Tablet PC is using Windows XP operation system.
Smart Phones	Windows Mobile-based Smart phones have all the features of a mobile phone, but also let you email, instant message, surf the Web, listen to music, etc.

Table 4.2 Devices description

Options	Server	Client	Connection
My first choice	Desktop PC	Tablet PC	Wireless LAN
Second	Desktop PC	Tablet PC	UMTS
Third	Desktop PC	PDA	GPRS

Table 4.3 Device options

4.3 Data Process and Storage

A test VRML version of the dataset (Alterra) has been made enabling implementations of VRML players, which can be used by mobile devices. A combination of database of services with VRML model will provide the location-based services to the tourists [16]. There will be a default 3D scene when the application started. When receiving the positions and orientation from the GPS hold by users when they are moving, the whole 3D model will send to the devices, and according to the orientation, the application decides which part of 3D will show up, as there are 12 directions belong to 12 different zones each of which has its own zone number. What the users can see is the selected direction of 3D scene in front of them the size of which is 20~50 meters in length and 30~50 meters in width. If the user is standing right on the boundary of two zones, the zone with smaller zone number has the priority. If the users keep stationary, the same position and orientation will generate even the users rotated at the same point which was proved in the discussion part in chapter 7. The difference between the concept and the design is that selection instead of clipping is happened because the whole model is loaded at the beginning actually and the application just selects the proper direction of 3D scene to display. The clipping can be the future work after this thesis as it involves some complex situations. The reason why we do not clip this time is that the database is quite small and simple so there is no need to do this. Moreover, not all the GPS data will cause the process of 3D model. Each time, at least 10 and maximum 12 GPS records was captured, the next step, the displaying part of the 3D, will then start. So in average, every 1~2 seconds the 3D model will refresh.

The height of the user's view can be changed interactively by selecting different values. It can make the users have a larger sight when the height is higher, for example 20 meters, or have a more reality landscape when the value is low, such as 1~2 meters, the range of a human height.

Objects information is provided by clicking the 3D objects on the client devices. Then the client will send the request to the server with the object ID. After this, the related information will response to the devices from server and sometimes information can automatically push to the users like pop-up information. The storage and process of objects information are all happened in the server side.

There are several alternatives how to process and store the 3D data. If the 3D data are stored on the server, processing could be fast if the server is powerful, managing and updating the data is convenient but the transmission of data to the client may time consuming in that the connection may not be stable all the time. The entire data can also stored on the client but it is not suitable for PDA because of its hardware limitations and it is also very hard to update if there are so many PDAs. 3D data process can be done in either server or client depending on the capacities of the devices and data trade-off. The options are listed in the table 4.4.

Options	3D data storage		3D data process	
	Server	Client	Server	Client
1		√		√
2	√			√
3	√		√	

Table 4.4 Varying options to store and process 3D data

I intend to design a client-server based system. So 3D data is stored in the server and the client will download it. Then the application will select the 3D model instead of clipping, and this process is done by the client device. So option 2 is my choice and option 3 could be the future work with the clipping process.

4.4 User Interface

Virtual Reality Modeling Language (VRML) is a scene description language for representing 3D interactive models on the web [8] and the X3D, which is the successor to VRML, is considerably more mature and refined standards than VRML [26] . But until now, the X3D browser is still under development. So the application can use VRML- based browser interface. But a terminal-based application is also a solution. Here is the comparison of the two:

Type of Application	Description
VRML-based browser	• Use the browser of the devices, so there is no software installation • Most process will be done on server side • Update only at the server • Little storage needed
Terminal-based application	• Specific software(e.g. Visual basic) development • Process can be done on client and server • Update needed for all the users • More storage space needed

Table 4.5 Comparison of Application

In the thesis work, the terminal-based application will be chose to develop the application as visual basic provides enough internal function for development.

4.5 Transmission

Network bandwidth limitation will be removed with third generation (3G) cellular system using Universal Mobile Telecommunication System (UMTS) [18]. UMTS is expected to offer a powerful environment for location based multimedia services [36]. The working distance of this system in our working condition is enough for the application. Other technologies like LAN, WiFi (WLAN), GPRS or Bluetooth can also be used. LAN has the merit of stable and fast connection but you need network cable which is not suitable for field work. We can use it to test whether the user interface, 3D data and information data is working properly. Wireless LAN can be used for navigation through a building, in combination with RFID. GPRS is an alternative for UMTS unless UMTS is available at this moment. There is a simple comparison of these three connections:

Connection	Description
UMTS	1920K bit/s data transfer rates
GPRS	Theoretical maximum speeds of up to 171.2K bit/s[37]
WLAN	Wireless Local Area Network, having average10M bit/s connection and stable

Table 4.6 Three transmission technique

XML or HTML [38] is transmitted between the client and server side. Wireless connections always have many factors which can influence the connections, in the application, this should be considered. In the thesis work, because the WLAN is available on the Tablet PC and the WLAN can work outside when the handheld is close to the wireless access node. Moreover, the testing area is just around the buildings where the wireless can work, so WLAN is the choice. But UMTS or GPRS is preferred in that they have no range restriction as WLAN.

4.6 Approach Phase

Depending on what has been discussed previously about the device, user interface, connection, data procession and storage and software. The approaching steps are shown as follows:

I. All the 3D data and objects information are stored and processed on the client devices, like the “Roman Limes” application [17]; GPS is not enabled. This phase is to see how the application works.

II. All the 3D data and objects information are stored and processed on the client devices; GPS is enabled. This phase is trying to pass the position and orientation to the 3D data then make the selection of data and finally display the 3D world.

III. Objects information is stored and processed on the server; 3D data is stored and processed on the client; Wireless LAN is used for connection; GPS is enabled. This phase is trying to make a links between client and server and transfer the necessary files.

IV. All the 3D data and objects information are stored on the server; 3D data is downloaded from the server then processed on the client; Wireless LAN is used for connection; GPS is enabled. This phase is to see how the 3D data is transmitted to the client.

V. All the 3D data and objects information are stored and processed on the server, then send the data to client by requests; Wireless LAN is used for connection; GPS is enabled. This is the final purpose of this thesis work.

VI. All the 3D data and objects information are stored and processed on the server, then send the data to client by requests; UMTS or GPRS is used for connection; GPS is enabled. This could be done when the PDA is powerful.

The steps can be described in the following table:

	3D data storage		**3D data process**		**information storage and process**		**connection**		**Position & Orientation**	
Steps	**Server**	**Client**	**Server**	**Client**	**Server**	**Client**	**UMTS or GPRS**	**Wireless LAN**	**GPS**	**No GPS**
I		√		√		√				√
II		√		√		√			√	
III		√		√	√			√	√	
IV	√			√	√			√	√	
V	√		√		√			√	√	
VI	√		√		√		√		√	

Table 4.7 Application development approach: subsequent steps

5. IMPLEMENTATION

In order to enable the design of the prototype application, it is necessary to consider the relation between functional design and actual implementation. In the first part, hardware and software are chosen. Then 3D model and the objects information database are properly created. Finally it will show the user interface which will connect the database in remote server, show the 3D model and information of user interested objects. At the end of each part, there will be a short description if there are some differences between the functional design and implementation. The implementation of the application follows the application approach steps in table 4.7.

5.1 Hardware and Software

To fulfill the application a Tablet PC is used. An Oasis Media Bluetooth GPS receiver will provide the position and orientation. The specifications are listed in the table 5.1. In de developing phase, a common used desktop PC is acting as a server which is power enough for the testing purpose. GPS data will be transmitted in NMEA format using Bluetooth technology from GPS receiver to application on the Tablet PC.

Tablet PC		GPS receiver	
Brand	HP	Brand	Oasis Media Bluetooth
CPU	Pentium® M 1.10GHz	Channels	16
Memory	512 Mb RAM	Features	Designed for urban canyon environment
Screen	1024*768	Precision accuracy	3 meters CEP (50%), without SA 7 meters (90%)
Wireless Connection	Intel® Pro/wireless LAN 2011 3B Mini & Bluetooth by HP	Baud rate	9600 bps
Operating system	Windows XP Tablet PC Edition 2005	Update rate	1 second
Display adapter	NVIDIA GeForce4 420 Go 32M	Interface	NMEA compatible

Table 5.1 Hardware specifications [39]

I didn’t choose the PDA during the implementation and testing. PDA is not so powerful to display 3D model while Tablet PC is suitable for running the application. A Tablet PC has enough storage space and is power enough. When the Tablet PC is used for testing, wireless LAN connection is the main connecting technique between the client and the server.

The software for developing is Visual Basic 6.0. It is used to make the user interface. IIS (internet information services) web server is used for the client-server solution. At the server side, ASP (Active Server Pages) is running inside the IIS, an ADO (Active Data Objects) connection is used to access the database on the server.

5.2 Position and Orientation controlled 3D Model

The 3D model below looks not highly realistic as only blocks are made. A realistic style is not chosen because the interface design is trying to generally reflect the landscape of the real world and be able to provide the right scene for a certain position and orientation. The blocks are clickable. When clicked, the application will send a request to the server asking relevant information about the object.

With the position and orientation, the selected 3D model appeared on the screen. The commands in the frame are examples of selecting the 3D scenes. Position specifies the user position in the VRML coordinate system. Orientation determines the direction in which the user is looking.

```
{   Position 101.405 10 -38.167
    Orientation 0 1 0 2.09
}
```

In the example, ‘0 1 0’ specifies a rotation along the Y axis in the anti-clockwise direction. As in the concept, the whole scene is divided into 12 zones, so the range of the angles is from -3.14 to 3.14 in VRML specification. The number 2.09 is the rotation angle, which is formatted in radians in VRML. It indicates the view direction. The figure 5.2 shows how the selected scene looks like.

The 3D model is constructed by combining a picture of north part of Wageningen, extruding buildings, trees and some characters. The shape of those buildings is similar to those in the real world and to simplify the model, no textures are added to those buildings. A few trees are made directly in the VRML file with the same trunks and crowns. Some characters are made at different directions of several certain points. Building, trees and characters are all mentioned in the testing part. One main purpose of the test is to know whether the application get the right position and orientation by asking the number of trees or writing down the characters they saw.

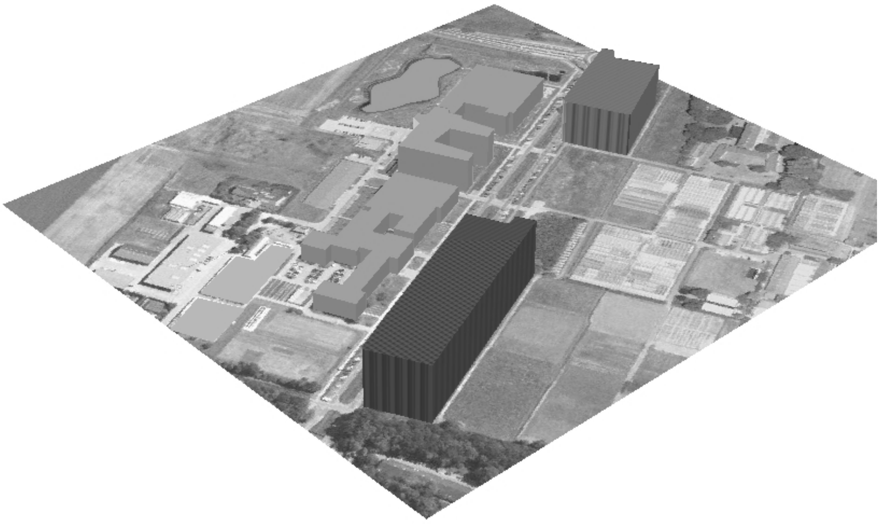


Figure 5.1 Three dimension scene of Alterra

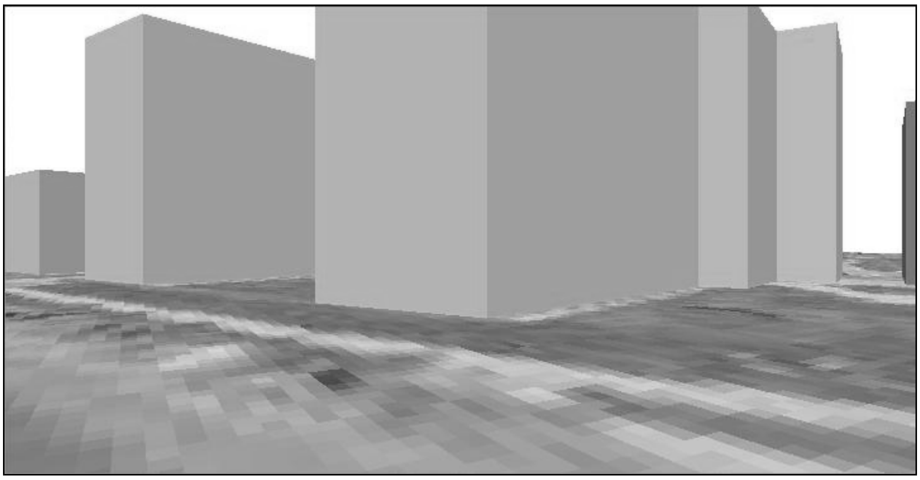


Figure 5.2 Selected 3D scene

5.3 Database

The table 5.2 is part of the simple database stored on the server. This table includes three attributes: ObjID, Name and Description. Figure 5.3 shows the basic four procedures of sending requesting from client and getting response from server.
When the server gets the request, an ADO connection (step 1) is created, and opens the database. Next it opens record set (step 2). The user will see the descriptions extracted from the 'Description' field in the selected record by specifying the ObjID or name (step 3 & 4).

Buildings		
ObjID	Name	Description
1	Alterra	Alterra in Wagenigen. New name: Gaia
2	West Alterra	Environment building on the west of Alterra
3	East Alterra	Library and Canteen. New name: LUMEN

Table 5.2 Simple Database on the Server

Here is the ADO connection and open scripts:

```
<%
   set conn=Server.CreateObject("ADODB.Connection")
   conn.Provider="Microsoft.Jet.OLEDB.4.0"
   conn.Open "c:/webdata/userinterestedInfo.mdb"
   set rs=Server.CreateObject("ADODB.recordset")
   rs.Open "buildings", conn
%>
```

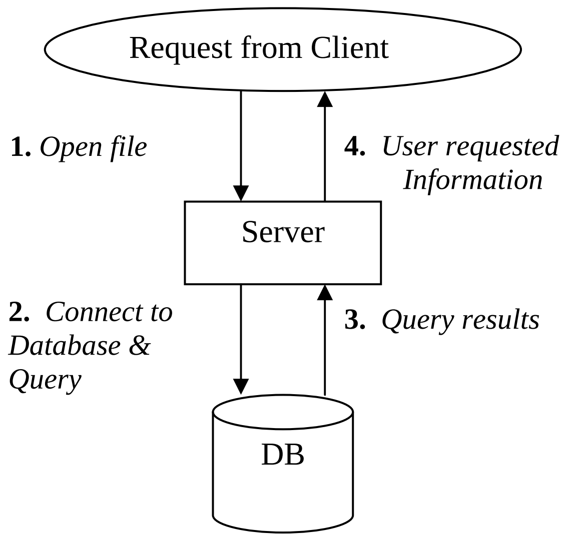


Figure 5.3 Querying from database

5.4 User Interface

The application interface is implemented in Visual basic environment to simplify the interface design by dragging the components, for instance, buttons labels, instead of writing the code. And it also has its own component for internet connection.

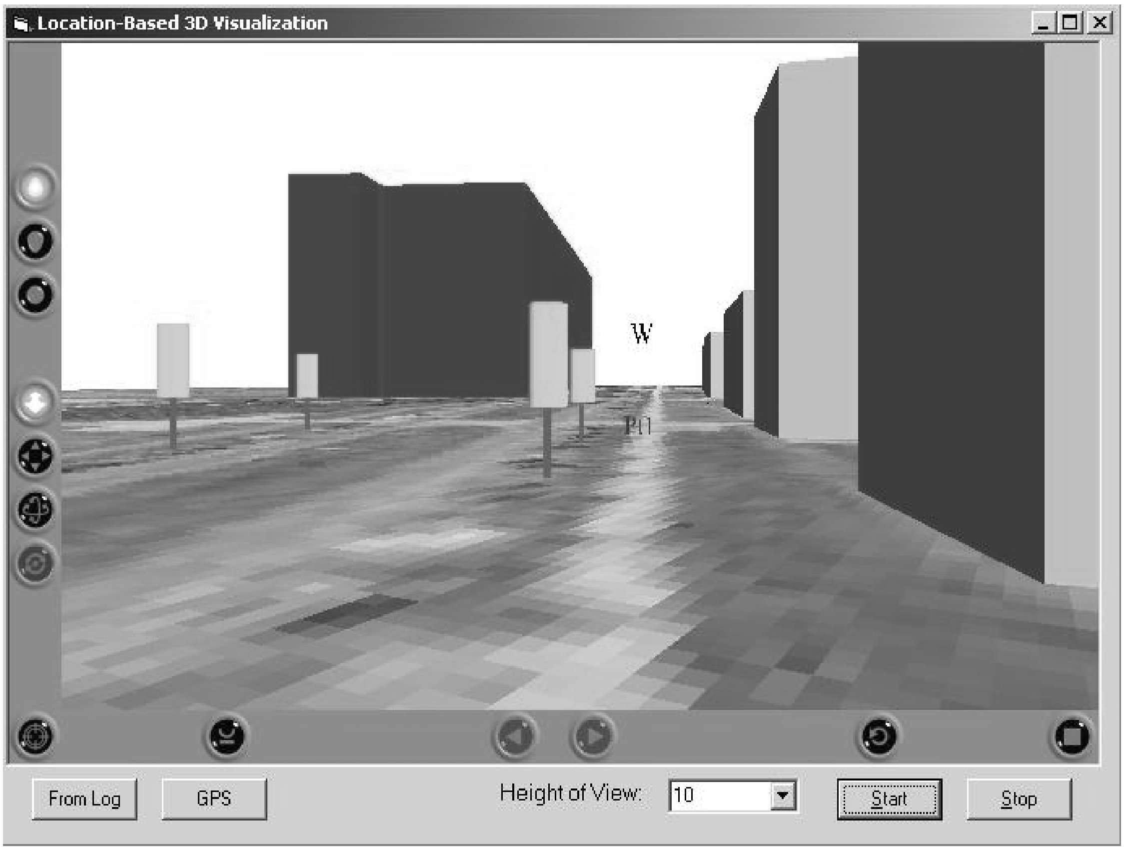


Figure 5.4 User interface

After downloading the 3D model into the application folder, in the user interface, as figure 5.4 shows, it includes the main frame where the 3D model will be displayed. Four buttons are located at the bottom. When the 'From Log' button is clicked, a form like figure 5.5 will appear. By clicking the button 'Read sentence from log', the application will read the GPS data from the log file where the position and orientation data is stored. Finally, a selected 3D model will show up in the main screen. The button is used for testing and if the users had recorded their tracks when they were moving. This function can let them see their footprints again.

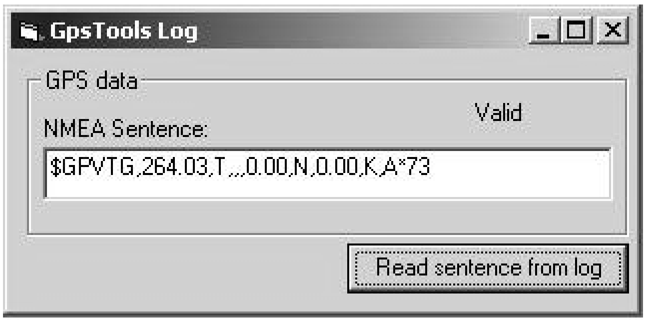


Figure 5.5 Navigating by Log files

The figure 5.6 is the form after clicking the button 'GPS'. This module will connect the GPS, and show the received GPS data in an understandable way. At the right bottom, if the 'save to log file' is checked, all the GPS data will be recorded and stored in a log file. So users can use the 'from log' button to see their route as it was told in the previous paragraph.

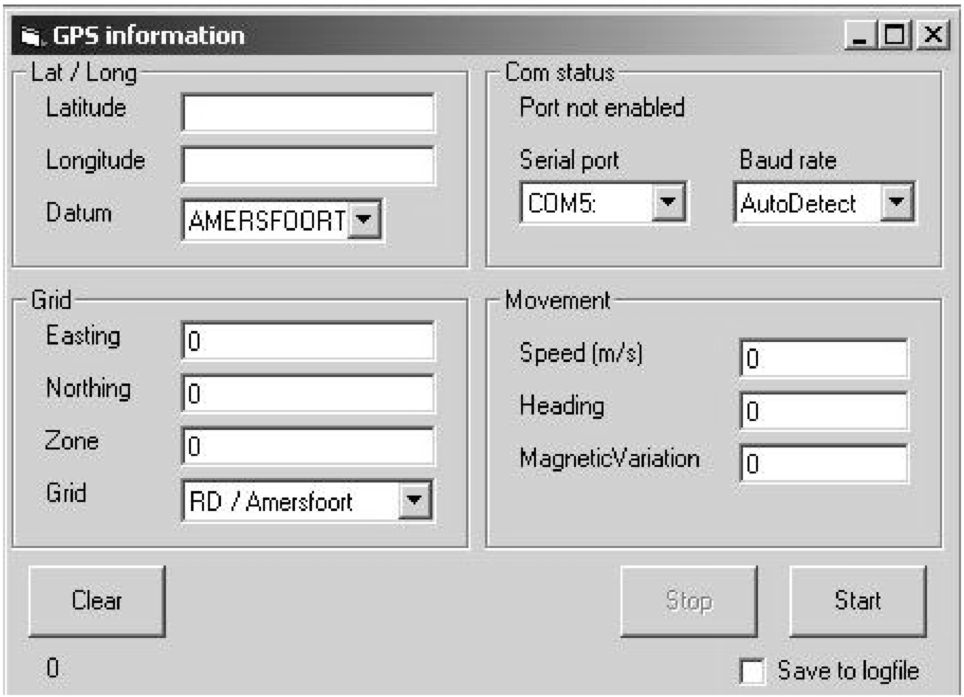


Figure 5.6 GPS component

There is a combo box at the right bottom of the main screen (figure 5.4). Users can choose the height of the viewpoint or type the number in the box. The value in that box sets the eye's level when the 3D model is displaying.

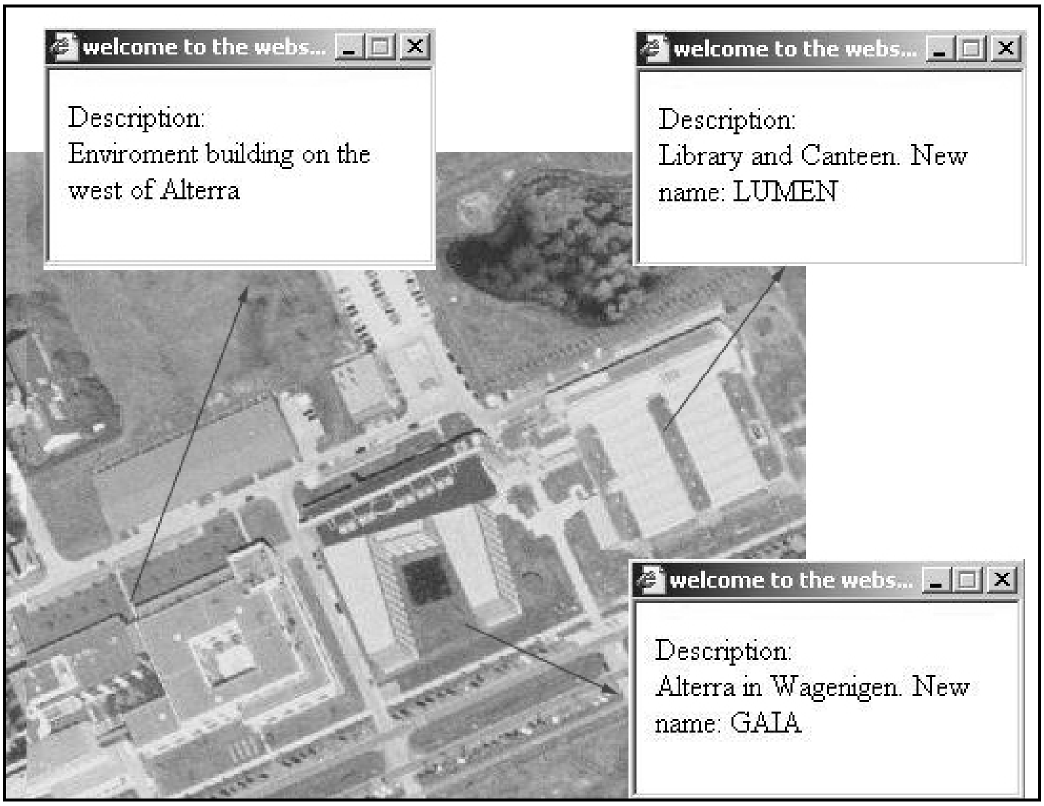


Figure 5.7 Object information

When users want to know more about the buildings, they can simply click on the 3D objects. Then a small web page will appear which describe the objects like the figure 5.7 showing.

6. TESTING

The application has been tested in the field near the city of Wageningen, around Alterra - de Haaf area. A simple 3D model has been constructed for this area (see figure 5.1). Before the field test, a preliminary user study was done with the Tablet PC in the office. This pre-test intended to let the users know how the application works without the GPS support, all the position and orientation data are read from a file. The pre-test can be done indoors or outdoors in a few minutes. The testing users are master students of geo-information science and some other users without GIS background. The number of users taking part in this test is 10.

6.1 Usability Test

The purpose of the field test is mainly to test the positioning and orientation functions in the 3D scenes and how the information services works. Transmission time is also considered because nobody wants to wait a long time for refreshing the new 3D scene. The whole evaluation was divided into two aspects: Usability of the application and usability of devices. In the first aspect, the testing is mainly about the position and orientation; information services and 3D model. The second is the comparison of the Tablet PC and PDA. In each aspect some questions were asked.

- **Usability of the application.**

The users were asked to walk around the Alterra building by a suggested pathway (the red and pink colored objects in figure 6.1). The bold red line in Figure 6.3 is the route. The path in the real world is the pavement around those buildings.

Users started the application, downloaded the 3D data, and connected the GPS. Here the downloading procedure only needs to be done by the first tester. The other testers don’t have to do it because it will remain in the application folder. They can start at the blue dot in the figure walking through points 1, 2 and 3. They can also start at each other place where the GPS can connect. However there is no priority to visit the three points which means that they can go to point 2 first or point 3, whatever they want.

During their walking, they tried to match the landscape in the real world and the 3D model on the computer screen at different locations with their walking direction. There are some letters put close to some buildings in the 3D model. At the particular points, as indicated number 1, 2 and 3 in the figure, the testers could find, by the proper orientation, those letters. This assignment is to test whether the application can get the correct information about position and orientation or not. Otherwise, the testers can not find those letters or they will find other letters instead of the expected and thus correct ones. When they requested some other information about the landscape, the server responded in a webpage format.

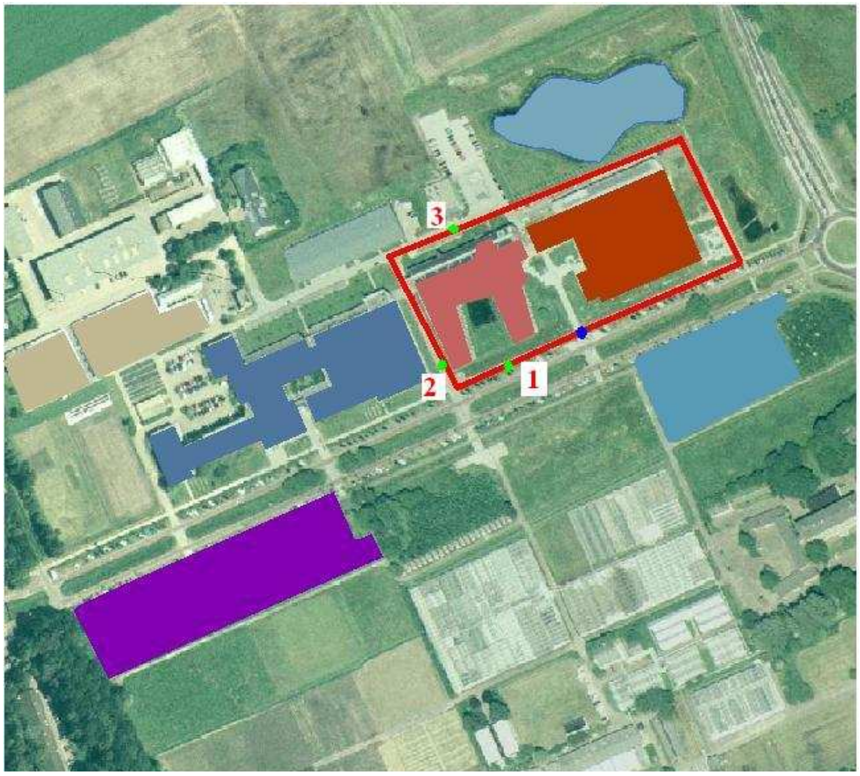


Figure 6.1 Simulate PDA sized application

The users will answer some questions. Question one to four are answered during the walking while the others are done afterwards. The questions to be answered in the field are:

1. If you are on the blue point (screen) and you walk a short distance to west (reality) then you see on the screen the abstract representation of the trees. How many trees do you see on the screen?

2. If you are on point 1 (screen) and you walk a short distance to proximately north (reality) then you see on the screen the abstract representation of a building. What letter do you read? Please write it down.

3. If you are on point 2 (screen), you walk a short distance proximately north (reality) if you are from point 1; look proximately south realities if you are from point 3; then you see on the screen the abstract representation of a building. What letter do you read? Please write it down.

4. If you are on point 3 (screen) and you walk a short distance to proximately south (reality) then you see on the screen which is the abstract representation of the building. What letter do you find? Please write it down.

Afterwards they have to answer the following questions:

5. How many buildings did you see during your walk on the screen?

6. How many characters did you see during your walking on the screen?

7. Did you find the provided object information?

8. Did you mainly move on bases of the real world or mainly base on the 3D scene at the screen?

The answer to those questions below has 5 scales: 1—No use; 2—Need improvement; 3—Good; 4—Satisfactory; 5—Very Good.

9. Does the landscape in the application topologically match the real world?

10. Are you satisfied by the refresh speed of the application?

11. Could you easily orient yourself in the 3D scene?

- **Usability of the devices.**

The device is a Tablet PC with a Bluetooth GPS receiver (Figure 6.2). As the application will only be working on a PC, it is not possible to test it on a PDA. However, a smaller interface of the application can be created by changing the screen size to one that equals to the size of a PDA (Figure 6.3).

Figure 6.2 Tablet PC and GPS receiver

So the 3D model will become smaller but the model itself is the same as the one on the full screen. During this part of the testing the users have to be aware that they are not holding a PDA but using a PDA simulator instead.

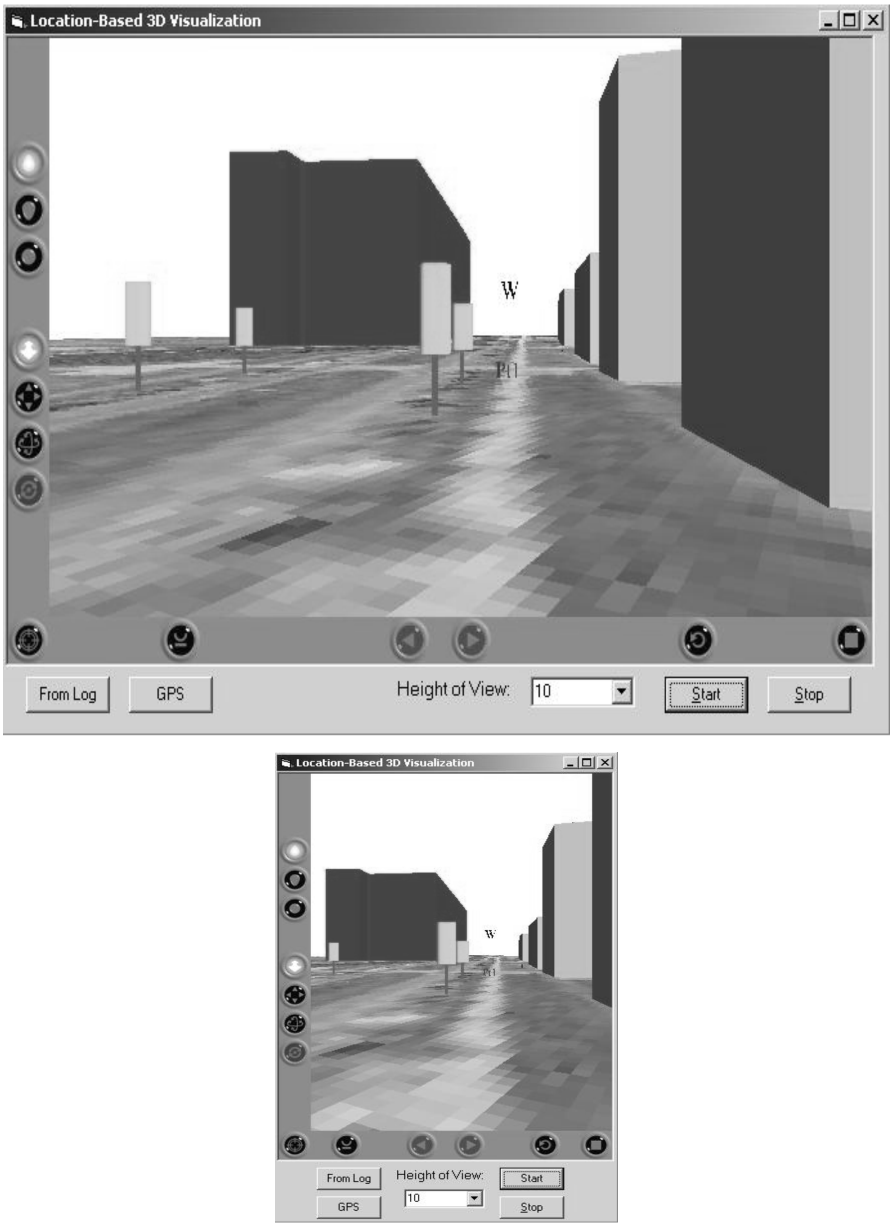


Figure 6.3 Simulated PDA- sized application

The tester is only using a Tablet PC both to show the application at its real size, and to show a view how it would look on a PDA device. The 3D model they will see is the same. The question is:

12. Can you still see the trees and letters on PDA screen size?

6.2 Test results

The ten testers are some geo-information science master students and some students without any GIS knowledge. From the table 6.1, the question numbers are corresponded to the questions in section 6.1, the testing questions 1 to 4 are mainly to see whether the position and orientation works well or not. The expected answers to the first four questions are: 4, H, A and N. According to the answers to the first four questions, all the testers can see the trees and find some letters in the 3D model at the correct position and proper orientation like the picture showed in Figure 6.3. Sometimes they can see all the trees and the each entire letter and sometimes, if the orientation does not reflect the walking direction very well or the eye's level is too high or too low, only part of trees or letters can be seen on the screen. But sometimes they can not see any trees or letters.

Ques.	TESTERS										Correct Answer
	1	2	3	4	5	6	7	8	9	10	
1	4	4	4	3	1	4	3	3	4	3	**4**
2	H	H	H	H	None	H	H	None	H	H	**H**
3	piece of A	A	A	A	A	None	A	None	A	A	**A**
4	N	small piece of N	None	S	N	N	N	N	N	N	**N**
5	5	2	3	4	5	4	3	3	5	3	**6**
6	4	2	2	2	4	3	4	4	3	4	**5**
7	Yes	Yes	No	No	No	No	No	No	No	No	
8	Real	3D	3D and Real	3D	Real	Real	3D	Real	3D	Real	
12	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
Questions below: 1—No use; 2—Need improvement; 3—Satisfactory; 4—Good; 5—Very Good											
9	2	3	2	2	2	2	2	2	2	3	
10	2	4	1	3	2	2	3	2	3	4	
11	2	4	2	2	2	2	2	2	3	4	

Table 6.1 Test results

The reason why this happened is mainly because of the GPS data. It needs some time to catch the testers' walking direction. Testers have to walk a distance like 2 to 10 meters then the correct orientation data can be got but at that moment they may already miss the trees or letters in the 3D scene. Testers really need walk to let the GPS catch the right directions. Sometimes they don't move or move too slow which means that they could not see the trees or letters.

According to question 5 and 6, most of the users can not remember how many letters and buildings they saw on the screen during their walking unless they are told to do. But all of them can remind the letters at the fixed point 1, 2 and 3. The testers were more focused on the 3D model and the real landscape instead of remembering the number of trees or letters. Some of them were moving on the bases of real world and some are using 3D model by checking the answers of question 8. Only two persons found the object information by clicking on the buildings. Testers suggested that an introduction for this function is necessary otherwise they were not aware of that. All of them are sure that they can see the same thing on a PDA-sized screen according to question 12.

All testers asked for more detailed 3D model because they found it is not very easy to orient them in the 3D scenes. They want to see the almost the same buildings in the 3D as that in the real world. And they hope the refresh speed can be improved by catching their speed. Moreover, if there is a sound that can remind them what to do, the testing would be easier in that they don't have to find the points on the screen but just hear the sound from the application.

7. CONCLUSION, DISCUSSION and RECOMMENDATION

The research questions are answered in the conclusion section based on the theoretical literature studies and testing results. In the discussion part, the delay of capturing the correct orientation degree is mainly discussed. Finally, further research work is suggested in the recommendation part.

7.1 Conclusion

In the thesis work, a 3D model info application was described and made for mobile users. It enables to see virtually the 3D landscapes according to the proper position and orientation in the real world. Objects information could be provided by accessing to the server.

The thesis research is divided into two parts as the first part is more theoretical where the different communication and positioning technologies are compared. Then an introduction for 3D visualization and client-server mode are come to. All of those are considered to develop my concept for this location-based service (LBS). The second part is more practical one where the location-based 3D visualization service was implemented. My research questions are:

1. What components does this application consist of?
2. How to capture the position and orientation?
3. How to present the 3D model?
4. How the linked object information (texts, sound and animations, etc.) are offered?
5. How can the application, resulting from the previous research questions be tested to assess its usability?

The questions 1 to 4 are more related to the theory and they are implemented in practical part. The last question is practical which are related to the testing part in the thesis.

The application is consisted of location detection, 3D model presentation and information support. GPS receiver is preferred, as compared with other network based position because it is accurate and easy to get. Although it can only work outdoors, the purpose is for using outside the buildings. With the GPS data, position and orientation information can be extracted by reading the raw GPS records using the

NMEA-0183 standard protocol. Most GPS receivers are capable to provide this standard format. The 3D model is constructed by extrusion of the polygons, lines or points and it was downloaded at the beginning of the application. Then it is presented by the control of position and orientation. The objects information in the 3D model is accessible on the server under the requirement of user clicks. Only texts information is provided in the application.

The usability testing shows that the position and orientation work well which shows the 3D scenes in front of the testers on the screen. But the orientation data extracted from GPS is not very fast so the delay makes the 3D model and real landscape not match so well all the time. More complex 3D model with textures is preferred and it would be nice if the PDA is power enough to run the application.

7.2 Discussion

In the test results, sometimes it took a few meters to get the right orientation from GPS and could even longer if the weather is not fine. We did a special test for the orientation following the pavement from the blue point to the point 2 indicated in the figure 6.1. The direction of this pavement is approximately southwest (around 240^{o}, 360^{o} means north). During the test, the variation of the orientation is recorded. And at certain points, we first rotated 360^{o} in clockwise direction and then reverse, finally record the orientation again.

Test Location	Coordinates (Dutch Grid)		Orientation (Degrees)		
	Easting	Northing	Orientation	After 360 degrees rotation clockwise	After 360 degrees rotation anti-clockwise
Blue point	174164	444231	154.71	154.71	154.71
4m west	174159	444229	234.67	234.67	234.67
25 m walking			**234.67~~274.17**		
Point 1	174117	444210	254.95	254.95	254.95
3m to North	174114	444214	352.56	352.56	352.56
15 m walking			**240.07~~261.18**		
Point 2	174082	444190	188.14	188.14	188.14
2m to West	174080	444189	223.19	223.19	223.19
4.5m to north	174083	444201	348.92	348.92	348.92

Table 7.1 Orientation discussion

Table 7.1 shows the results of the test. From the table, it could be seen that rotation at the same point has no influence on the orientation. Only movement will generate the orientation. We started from the blue point facing to the southeast direction, the GPS record has a 154.71^{o} orientation belonging to the zone 6(150^{o} ~180^{o}) as showed in figure 7.1. After 4 meters walking to west, the GPS provided the right orientation along the pavement. The similar situation happened in point 1 and 2, that after walking 2, 3 or 4.5 meters to the certain direction, orientation can be captured.

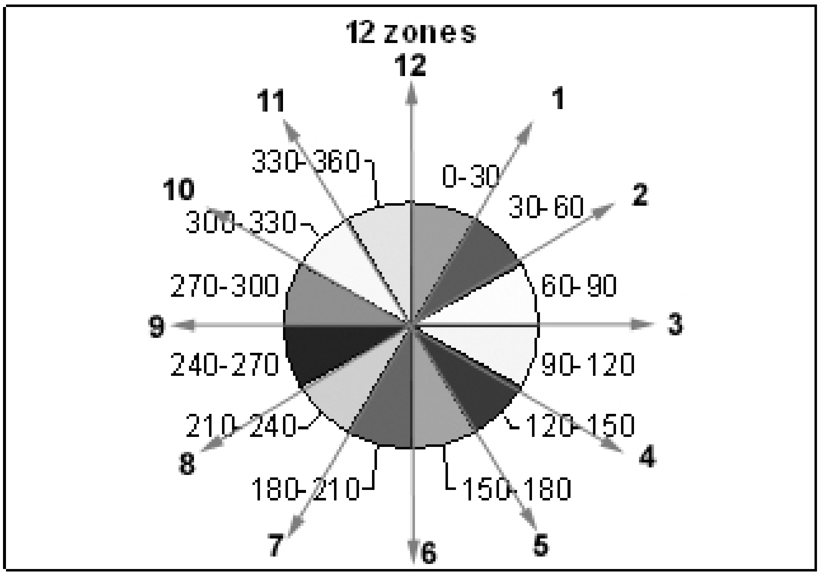


Figure 7.1 Twelve zones

Between the 25 meters way from blue point to point 1, 40^o variation was detected. 21^o variation was found when walking from point 1 to point 2. The reason is that small turning to a different direction will cause the change of the orientation data because we can not move constantly. If we equally divide the zones, the 3D scenes may not match because the orientation variation could cause the application to show the 3D scenes belong to neighborhood direction, although the 3D scenes have some overlaps as showed in figure 7.2. But there is no overlap between neighborhood zones. For example, if the expected direction is number 9 (240^o ~270^o in the figure 7.1), the application could show the direction of number 8 (210^o ~240^o) or number 10 (270^o ~300^o) when the variation of orientation is larger than 30^o.

Changing the range of each zone may partially solve this problem. In the original application, 12 zones are set and they equally divided. Each zone indicate one direction so one direction contains 30^o (360^o /12). Adjustment could be made by increasing the range of zone along the road and decrease the range of zone not along the road. In this case, the range of direction 9 will increase from 240^o ~270^o to 225^o~ 275^o while the range of zone 8 and 10 will decrease by 15^o and 5^o each. After the adjustment, the degree of orientation falls into 225^o ~275^o will show the same 3D scenes which belong to direction 9.

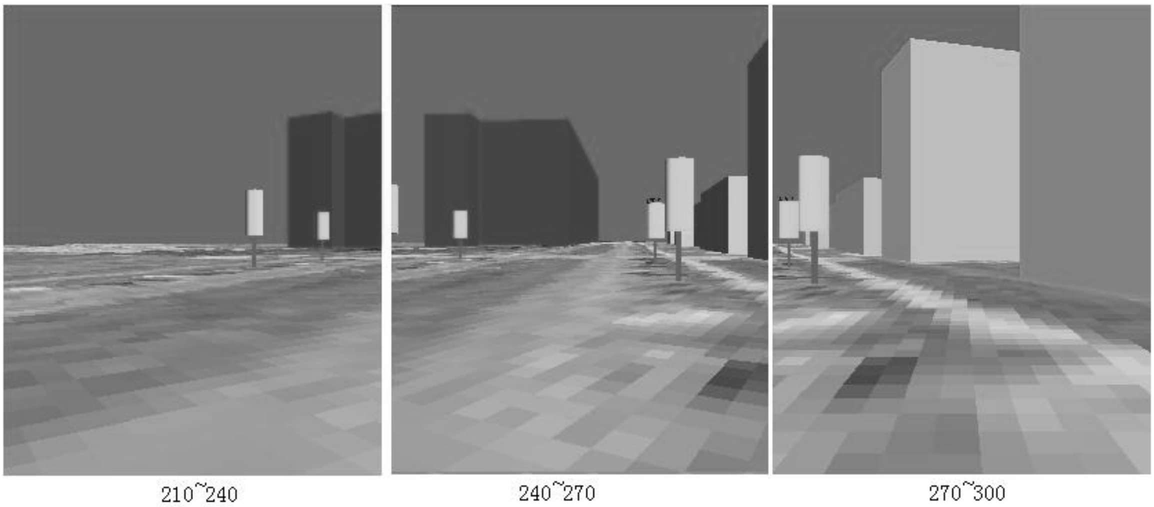


Figure 7.2 Overlaps between different directions

The 3D model in the application was made by ArcScene and converted into VRML. In the VRML files, there were some changes in order to make the objects accessible to the server for information. And there were also some changes like adding textures on the objects because some 3D symbols in ArcScene lost after the conversion to VRML. But matching the code in file with the objects was boring if the number of objects is large. So an automatically generation of 3D building model is needed. In future, X3D could also be a good choice to implement this option.

7.3 Recommendation

The application was mainly written by Visual Basic 6.0. Visual Studio is recommended for future development because more help files could be found on the internet and it also support visual basic specification. The objects information only contains very simple texts which tell what kind of building it is. Multimedia information, such as sound, video or animation can add into the application.

The speed of the users is not considered in the application. In the application, the view refreshed when 10~12 GPS records were received which means 1~2 seconds depending on the GPS receiver; a supposed speed of normal walking is used. In the future work, the speed can also be a parameter in the 3D visualization to control the refresh frequency. Lower speed may offer a more detailed 3D model with lower refresh speed. Higher speed, such as cycling or driving, may need faster refresh speed and simple 3D model could be enough.

There is no real clipping happened in the application because the dataset is small. The whole dataset was downloaded from the server side at the beginning of the application. If the dataset is quite big, clipping could be done at the server side especially when the client device is a PDA which is not power enough and has not enough storing space. With the proper position and orientation, the server can clip the big dataset and then send to the client side which will also reduce the transmission time. A VRML browser always gets the entire model. But some people made some extensions of VRML enabling progressive, on-demand downloading and smooth rendering. Guéziec et al. [40] have presented a framework for streaming geometry in VRML. And X3D could solve this problem. Referring to the approach steps in table 4.7, chapter 4.6. This thesis work stopped in step 4. Future work could include step 5 by processing the clips on the server and using UMTS for connection as indicated in step 6. And in the step 6, if the clipping function is implemented on server, devices could be PDA instead of Tablet PC. But adjustment in the application is required.

The application can only work outdoors because of the GPS receiver. If the position system can combine the GPS, Cell ID and RFID technologies, which can provide the location inside the buildings, the whole system can work both outdoors and indoors. Moreover, objects under the ground can also be built to provide more information.

REFERENCES

1. A.Tsalgatidou, Jari Veijalainen, Jouni Markkula, Artem Katasonov, Stathes Hadjiefthymiades, *Mobile E-Commerce and Location-Based Services: Technology and Requirements.* 2003.
2. J.Fagerberg, *Summary of THE STRUCTURE OF THE EUROPEAN LBS MARKET 2005.* 2005.
3. E.Kaasinen, *User needs for location-aware mobile services.* Personal and Ubiquitous Computing, 2003. **7**(1): p. 70-79.
4. R.v.d.Togt, E.Beinat, S.Zlatanova, Henk J. Scholten, *Location Interoperability Services for Medical Emergency Operations during Disasters.* 2005.
5. K.Laakso, O.Gjesdal, Jan Rasmus Sulebak, *Tourist information and navigation support by using 3D maps displayed on mobile devices.* 2003(Workshop "HCI in mobile Guides", Udine (Italy)).
6. PocketPC, 2. Microsoft Mobile Devices, *http://www.pocketpc.com/ (Accessed August 2005).*
7. Palm, *http://www.palm.com/ (Accessed August 2005).*
8. R.R.Lipman, *MOBILE 3D VISUALIZATION FOR CONSTRUCTION.* 2002.
9. J.Roth, *Flexible positioning for location based services.* IADIS International Journal on WWW/Internet, 2004. **1**: p. 18-32.
10. C.S. Jensen, A.Friis-Christensen, Torben B. Pedersen, D.Pfoser, S.Saltenis, N.Tryfona, *Location-Based Services - A Database Perspective.* 2001.
11. J.Korhonen, *Introduction to 3G Mobile Communications.* 2003: p. 5.
12. wikipedia, *Http://en.wikipedia.org/ (Accessed September 2005).*
13. C.Smith, D.Collins, *3G wireless networks.* 2001: p. 140.
14. P.Wentzel, R. v. Lammeren, M.Molendijk, S.de Bruin, A.Wagtendonk, *Using Mobile Technology to Enhance Students' Educational Experiences.* 2005.
15. d.Berg, N. v., *3D Visualization in location-based services.* Thesis report, Wageningen University, 2005.
16. L.Chittaro, S.Burigat, *Location-aware visualization of a 3D world to select tourist information on a mobile device.* 2004.
17. M.de Keijzer, Malgorzata Milosz, P.Naus, H.Zhou, *The Roman Limes-A location based service telling the story*. Course report, Wageningen University, 2005.
18. I.Rakkolainen, T.Vainio, *A 3D City Info for Mobile Users.* 2001.
19. I.Koeppel, *What are Location Services? – From a GIS Perspective.* ESRI Whitepaper, 2002.
20. K.Cheverst, G.Smith, *Exploring the notion of information push and pull with respect to the user intention and disruption.* 2001.
21. T.Halonen, J.Romero, J.Melero, *GSM, GPRS and EDGE Performance.* 2003.
22. UMTS world, *http://www.umtsworld.com/technology/overview.htm#a2 (Accessed November 2005).*

23. *http://www.uktelematicsonline.co.uk/html/wireless_wan_s.html (Accessed November 2005).*
24. B.Rao, L.Minakakis, *Evolution of mobile location-based services.* Communications of the ACM, 2003. **46**(12): p. 61-65.
25. R.v.d.Togt, E.Beinat, S.Zlatanova, Henk J. Scholten, *Location Interoperability Services for Medical Emergency Operations during Disasters.* 2005.
26. Web3DRepository, *http://www.web3d.org/x3d/specifications/#vrml97 (Accessed November 2005).*
27. Cortona, *http://www.parallelgraphics.com/products/cortona/ (Accessed November 2005).*
28. ISO/IEC14772-1, *Virtual Reality Modeling Language (VRML).* 1997.
29. ISO/IEC14772-2, *Virtual Reality Modeling Language (VRML).* 2004.
30. Data Export, *http://gis.washington.edu/esrm250/lessons/data_export/#vrml (Accessed November 2005).*
31. ISO/IEC19776-2, *Extensible 3D (X3D) encodings — Part 2: Classic VRML encoding.* 2005.
32. NMEA-0183, *http://www.kh-gps.de/nmea-faq.htm (Accessed November 2005).*
33. NMEA-0183, *http://www.nmea.org/pub/0183/index.html (Accessed November 2005).*
34. M.Kada, *Automatic generalization of 3D building models.* Spatial Information and Decision Making, 2002: p. 30-36.
35. O.Emem, F.Batuk, *GENERATING PRECISE AND ACCURATE 3D CITY MODELS USING PHOTOGRAMMETRIC DATA.* ISPRS congress, commission IV, 2004: p. 411.
36. G.Pospischil, H.Kunczier, A.Kuchar, *LoL@: a UMTS location based service.* Proc. of the International Symposion on 3G Infrastructure and Services, Athens, Greece, 2001.
37. GSM world, *http://www.gsmworld.com/technology/gprs/intro.shtml#1a (Accessed October 2005).*
38. R.Kromwijk, *THE USE GML IN LOCATION-BASED SERVICES.* Thesis report, Wageningen University, 2004.
39. GPS, *http://www.buygpsnow.com/item.aspx?itemid=385.*
40. A.Guéziec, G.Taubin, B.Horn, *A Framework for Streaming Geometry in VRML.* IEEE Computer Graphics and Applications, 1999. **19**(2): p. 68-78.
41. Visualbasiconlinehelp,h*ttp://msdn.microsoft.com/library/default.asp?url=/library/en-us/vbcon98/html/ (Accessed 2005-2006).*
42. ASP&ADO, *http://www.w3schools.com/ado/ado_intro.asp (Accessed November 2005).*

APPENDIX A SOFTWARE

1. Visual Basic 6.0

Microsoft Visual Basic, the fastest and easiest way to create applications for Microsoft Windows. Visual Basic provides you with a complete set of tools to simplify rapid application development [42]. Visual Basic has the tools for developing Data access application to access most popular database such as Microsoft SQL Server; Internet capabilities to provide access to documents, etc.

There is a windows CE toolkit for Visual basic 6.0 with which you can develop applications for a Palm-size PC using a subset of the Visual Basic language and ActiveX controls. The toolkit modifies the existing Visual Basic IDE to allow development for compact devices, but maintains a user interface that is consistent with the IDE used for desktop-application development. The Windows CE Toolkit consists of an integrated set of menus, tools, and toolbars that augment the desktop IDE. Application development for the Palm-size PC using the toolkit closely parallels application development for the desktop using Visual Basic 6.0.

2. ASP

ASP is a program that runs inside Internet Information Services (IIS) which comes as a free component with Windows 2000 or XP. An ASP file, with the file extension of .asp, is just the same as an HTML file. An ASP file can contain text, HTML, XML, and scripts. Scripts in an ASP file are executed on the server. The ASP file is just the same as the HTML file. The main difference between the execution of HTML and ASP is when a browser requests an HTML file, the server returns the file. When it is an ASP file, IIS passes the request to the ASP engine. The ASP engine reads the ASP file, line by line, and executes the scripts in the file and finally the ASP file is returned to the browser as plain HTML. The scripts are server scripts surrounded by the delimiters <% and %>. The sever scripts can be VB script or Java script including any variables, functions, procedures or operators valid. A single example is showed below:

```
<html>
<body>
```

```
<%   response.write("<h2>use HTML tags</h2>")  %>
<%   response.write("<p style='color:#0000ff'>This text is styled!</p>")  %>
</body>
</html>
```

An ASP file normally contains HTML tags, just like an HTML file. However, an ASP file can also contain server scripts, surrounded by the delimiters <% and %>. Server scripts are executed on the server, and can contain any expressions, statements, procedures, or operators valid for the scripting language you prefer to use.

By using the ASP, we can dynamically edit, change or add any content of a web page, respond to user queries or data submitted from HTML forms, access any data or databases and return the results to a browser, customize a web page to make it more useful for individual users, provides security in that your ASP code can not be viewed from the browser. Since ASP files are returned as plain HTML, they can be viewed in any browser, and minimize the network traffic if ASP programming is well done.

3. ADO

ADO is a Microsoft Active-X component which is automatically installed with Microsoft IIS. It is a programming interface to access data in a database. The common way to access a database from inside an ASP page is [42]:

1) Create an ADO connection to a database;
2) Open the database connection;
3) Create an ADO record set;
4) Open the record set;
5) Extract the data you need from the record set;
6) Close the record set;
7) Close the connection;

APPENDIX B ACRONYMS and DEFINITIONS

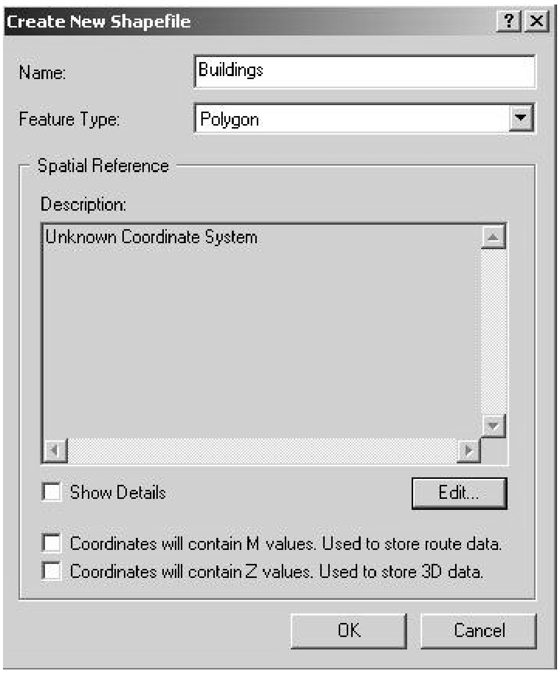
.

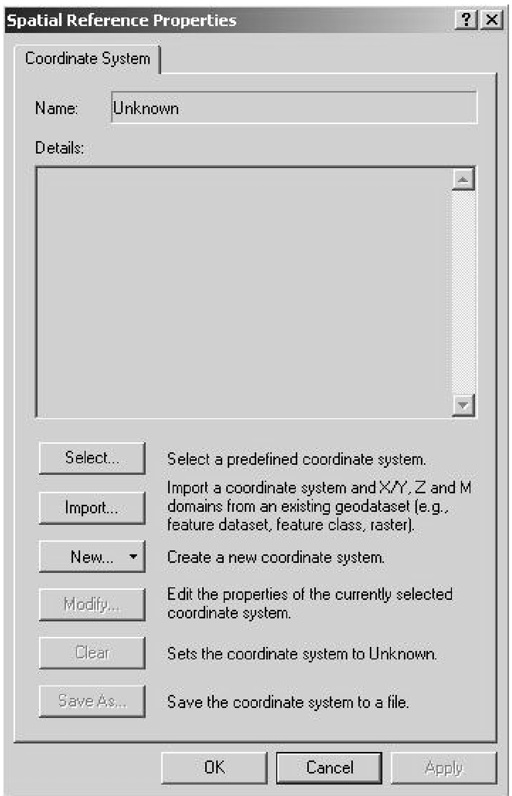
ADO	**A**ctiveX **D**ata **O**bjects
ASP	**A**ctive **S**erver **P**ages (Microsoft); server-side technology for dynamically-generated web pages. An ASP file can contain text, HTML tags and scripts. Scripts in an ASP file are executed on the server.
CDMA	**C**ode **D**ivision **M**ultiple **A**ccess is a form of multiplexing (not a modulation scheme) and a method of multiple access that does not divide up the channel by time or frequency, but instead encodes data with a certain code associated with a channel and uses the constructive interference properties of the signal medium to perform the multiplexing.
CDMA2000 =2.5G	a family of third-generation (3G) mobile telecommunications standards that use CDMA, a multiple access scheme for digital radio, to send voice, data, and signaling data (such as a dialed telephone number) between mobile phones and cell sites. It is the second generation of CDMA digital cellular.
EDGE =3G	**E**nhanced **D**ata rates for **G**SM **E**volution. A digital mobile phone technology which acts as a bolt-on enhancement to 2G and 2.5G General Packet Radio Service (GPRS) networks. This technology works in TDMA and GSM networks. EDGE (also known as EGPRS) is a superset to GPRS and can function on any network with GPRS deployed on it, provided the carrier implements the necessary upgrades.
GPRS	**G**eneral **P**acket **R**adio **S**ervice; a mobile communication technology designed to transport data.
GPS	**G**lobal **P**ositioning **S**ystem.
GSM =2G	**G**lobal **S**ystem for **M**obile communications or Group Special Mobiles; standard for mobile phones.
NMEA-0183	A standard for position data communication, for instance between a GPS receiver and a PDA, developed by the National Marine Electronics Association.
PDA	**P**ersonal **D**igital **A**ssistant; a type of handheld computer.
Pocket PC	A special version of Microsoft’s Windows Compact Edition (Windows CE) enabled with cellular phone and text messaging features. There are three versions of Pocket PC 2000, 2002 and 2003.

UMTS =3G	**U**niversal **M**obile **T**elecommunications **S**ystem. One of the third-generation (3G) mobile phone technologies. It uses W-CDMA as the underlying standard, is standardized by the 3GPP, and represents the European/Japanese answer to the ITU IMT-2000 requirements for 3G Cellular radio systems.
VRML	**V**irtual **R**eality **M**odeling **L**anguage
WCDMA	**W**ideband **C**ode **D**ivision **M**ultiple **A**ccess. A type of 3G cellular network. W-CDMA is the technology behind the 3G UMTS standard and is allied with the 2G GSM standard.
X3D	The successor of VRML, ISO standard defined by the Web 3D Consortium

APPENDIX C STEPS TO APPLY A NEW DATSET INTO THE APPLICATION

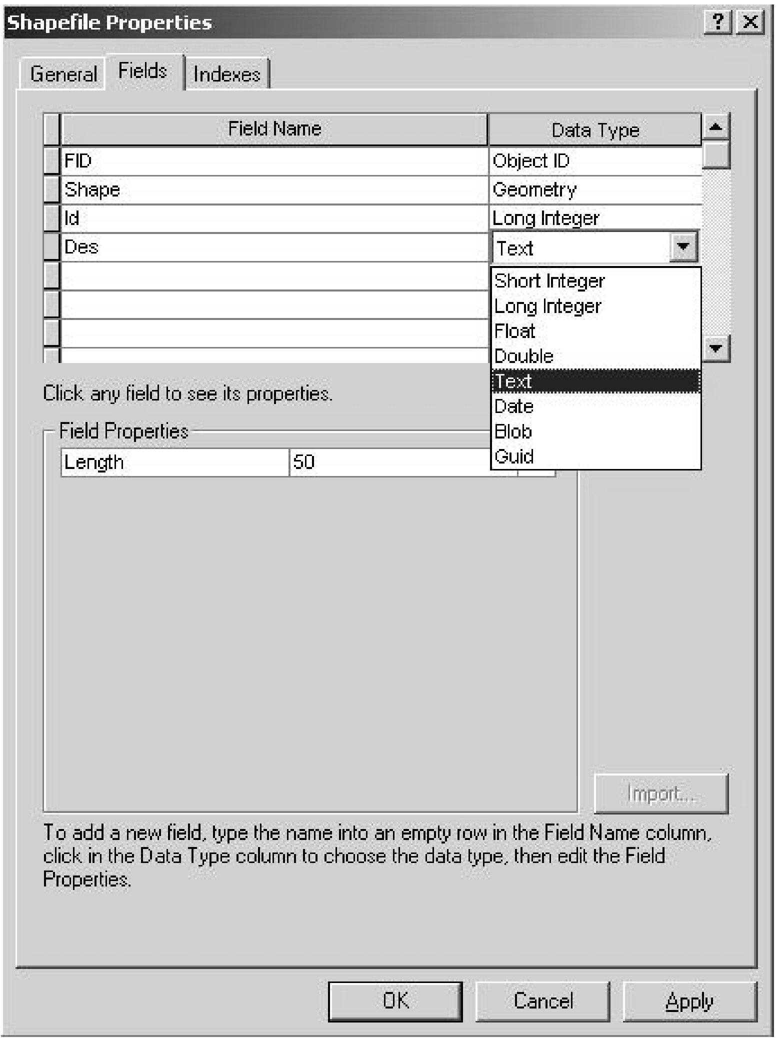
1. Unzip the zip file **visualization.zip**. You can put it anywhere in your computer. Here we unzip to C:\Temp\. There are three folders inside: **3Dvisualization** which is the application, **visual3D** where you can put all you temporary data, **visual_serverfiles** which contains the proper server computer.
2. Open ArcCatalog, browse to folder **C:\Temp\visualization\visual3D**. If you have different folder, then go to your own place.
3. Create a new shapefile by right clicking on the folder (here is "visual3D"); choose **New** then point to **shapefile**, open the "Create New shapefile" dialog, write down the name e.g. "buildings", choose **Polygon** as the feature type; Click **Edit…** to open the "Spatial Reference Properties" dialog, click **Import...** then browse to find the dataset which has the right coordinate system, click **Add**, then press **Ok**. Press **Ok** again to finish the procedure.



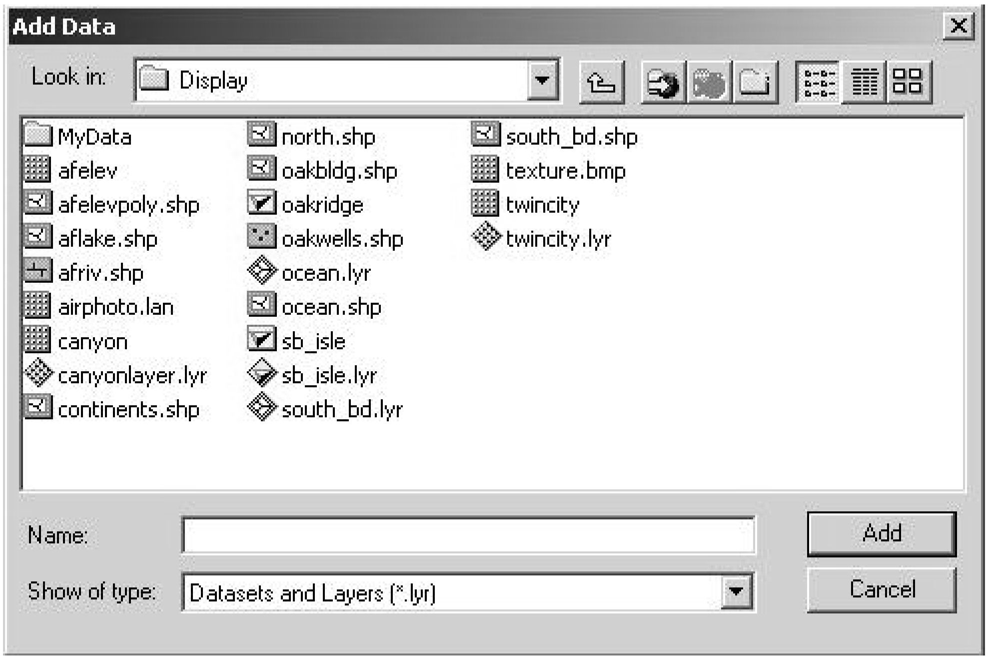


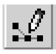
4. Right click on the new shapefile (name "buildings"), choose **Properties** to open the "Shapefile Properties" dialog, choose the **Field**s tab. Under the field name, click on the blank line and write down the name of a new attribute such

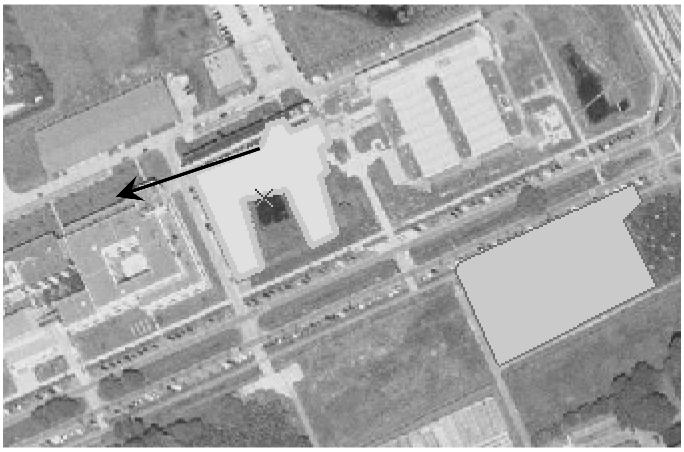
as "Description" and then choose **Text** as data type. Press **Ok** to close the dialog. This step is actually optional. If there are not too many buildings or other objects you want to construct, skip this step. Otherwise, this new attribute help you easily remind what this object is about.



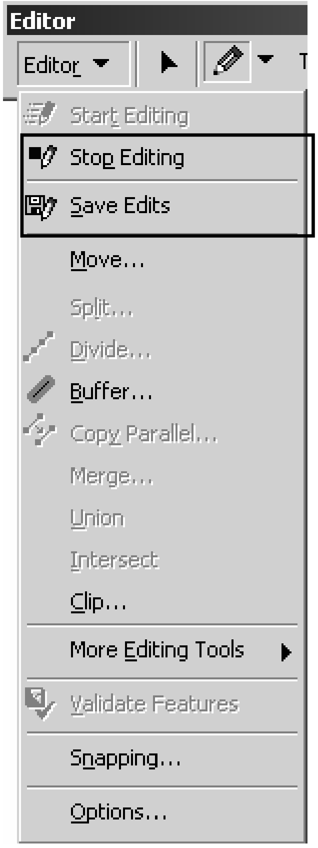
5. Repeat step 3 to create more polygon layers, each of which will represent one building. This time we skip step 4 because the 3D model we will construct is simple. Let's assume the name for those layers are building_1, building_2, building_3, etc.
6. Close ArcCatalog.
7. Open ArcMap to start a new empty map, add those buildings layer through "Add data" dialog. Also add a satellite image into the table of contents which is easier for you to identify some objects; Make sure the image also have the same coordinate systems as the layers. Click **Add** to close this dialog, and you will see the layers in the Table of Contents. Turn on the satellite image.



8. Click the  button, the **Editor toolbar** will appear; Click on the Editor then choose **Start Editing,** set the task as “Create New Feature” and the target layer is “buildings” or others if you defined a different name; Use the sketch tool  to draw a polygon around a building based on the satellite image; the figure below shows an example which is the polygon of Alterra Gaia building in Wageningen.

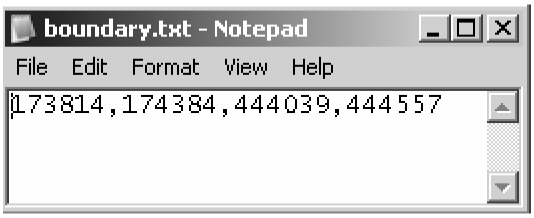


You can delete this polygon if you don’t satisfy and then redraw it again. Click Editor then choose **Save Edits.** If you think you have finished, choose **Stop Editing** in the Editor Toolbar**.**



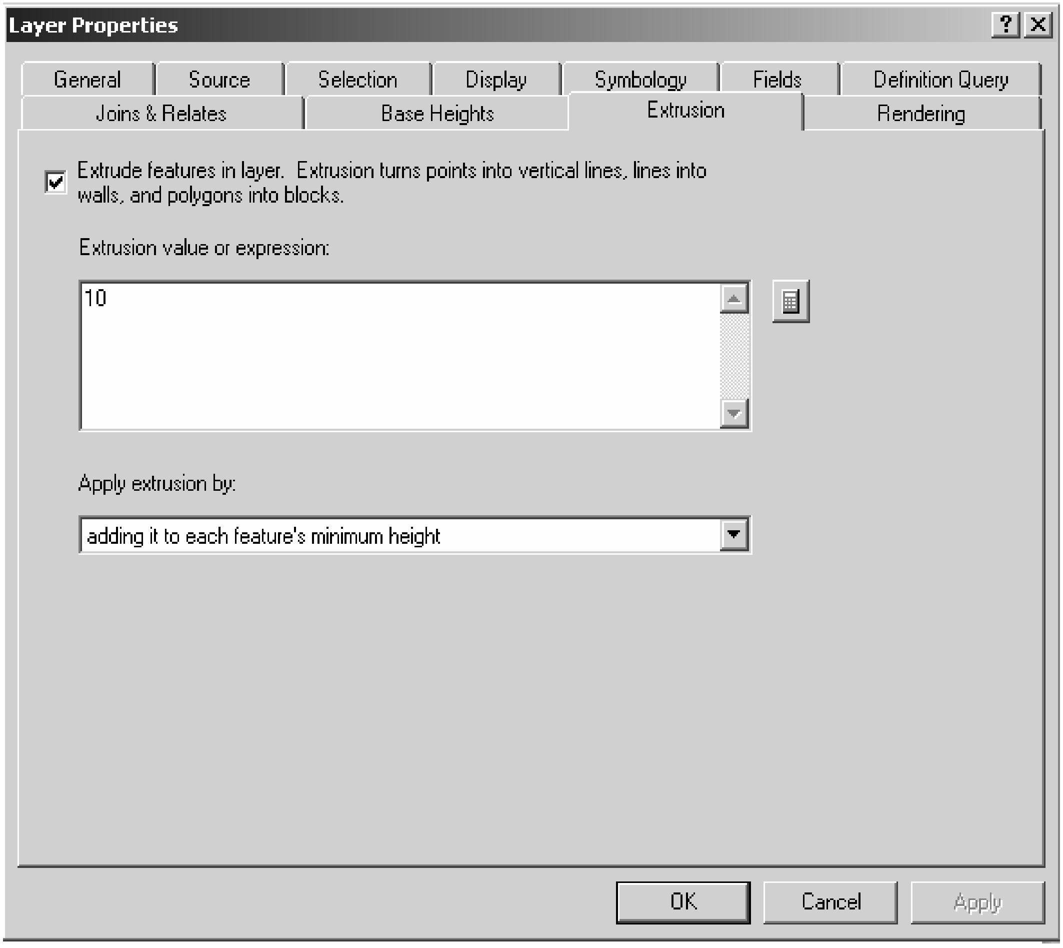
9. Repeat step 8 to edit more polygons to represent the buildings or other objects. Save this file from **File -> Save**. Write a name for this file, e.g. objects_3D.mxd.

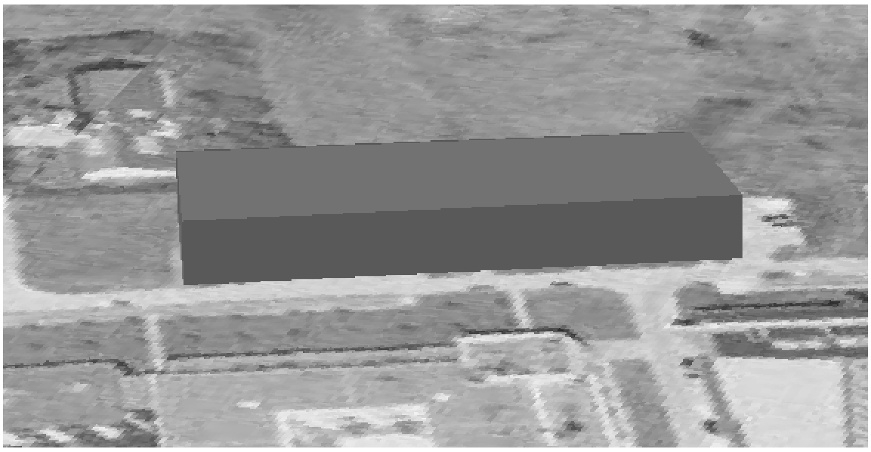
10. Click the  button to find the Xmin, Xmax, Ymin, Ymax coordinates in your study area, write "**xmin, xmax, ymin, ymax**" in a text file name boundary.txt. The comma between them is required. Save the file into **C:\Temp\visualization\3Dvisulization**. Close ArcMap.



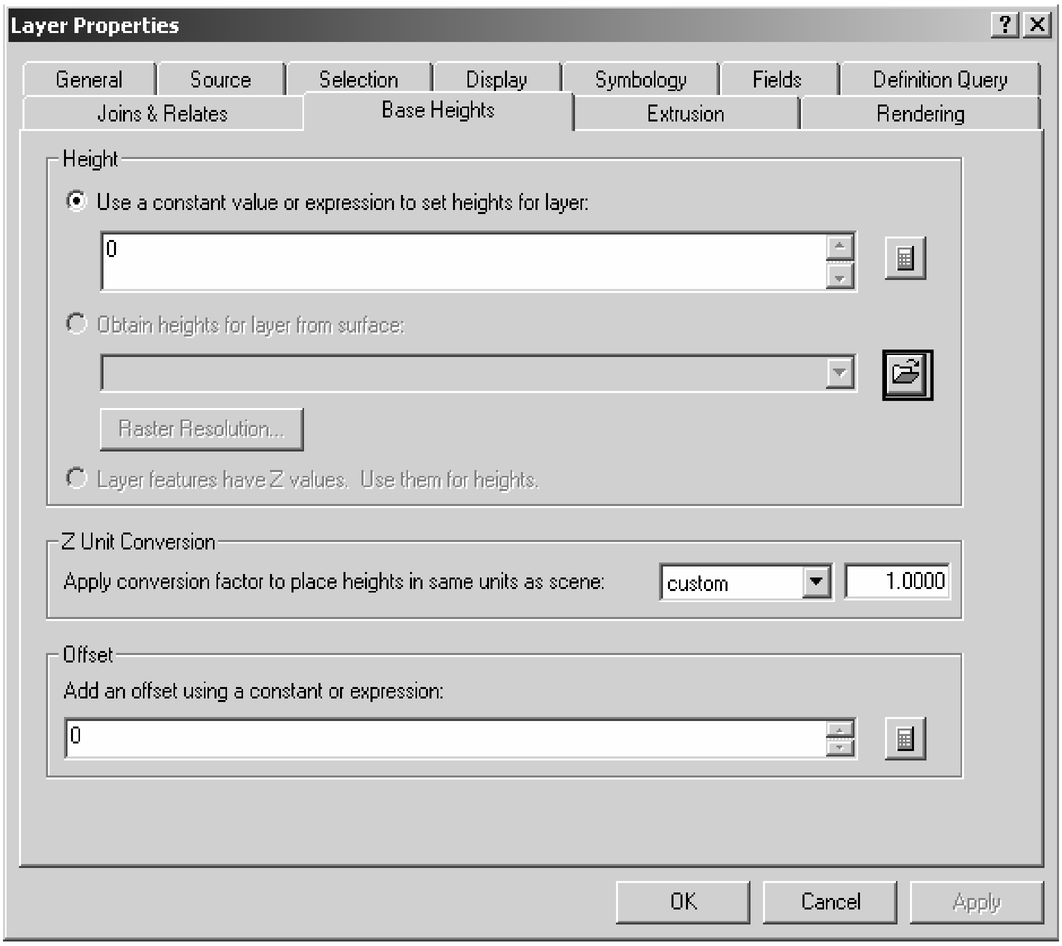
11. Open ArcScene. Add those Building layer and the satellite image through "Add Data**"** dialog.

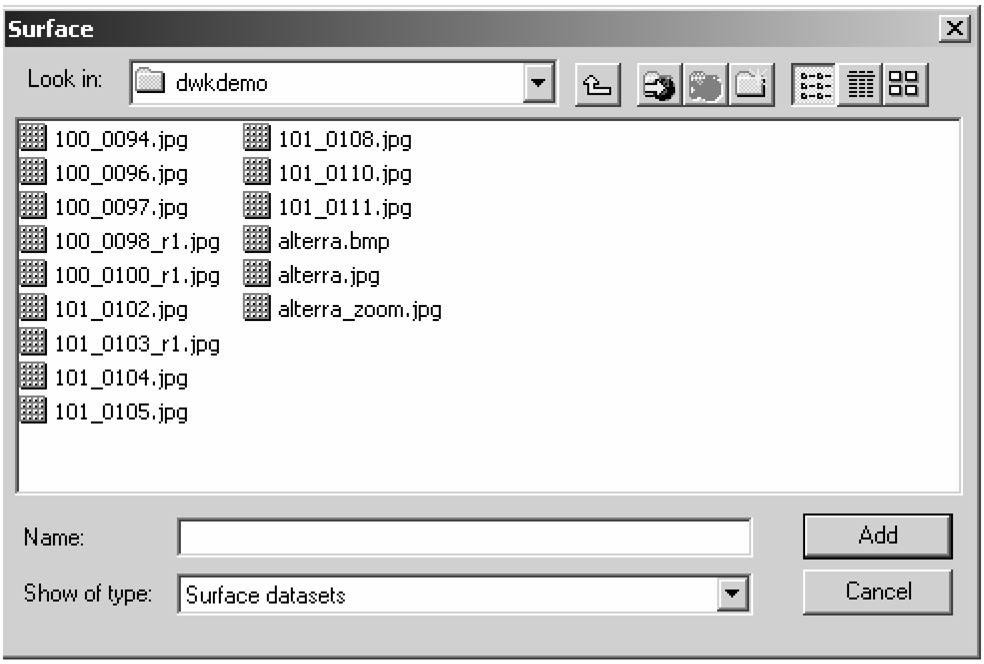
12. Open the "layer Properties" dialog of one of the Buildings layers (e.g. building_1). Choose the "Extrusion" tab. Input a number in the "Extrusion value or expression:" box, we type 10 here. Leave the apply extrusion by as "add it to each feature's minimum height". Click **Apply**. Drag this dialog so you can see the result. The color of the building can be different and you can change it;



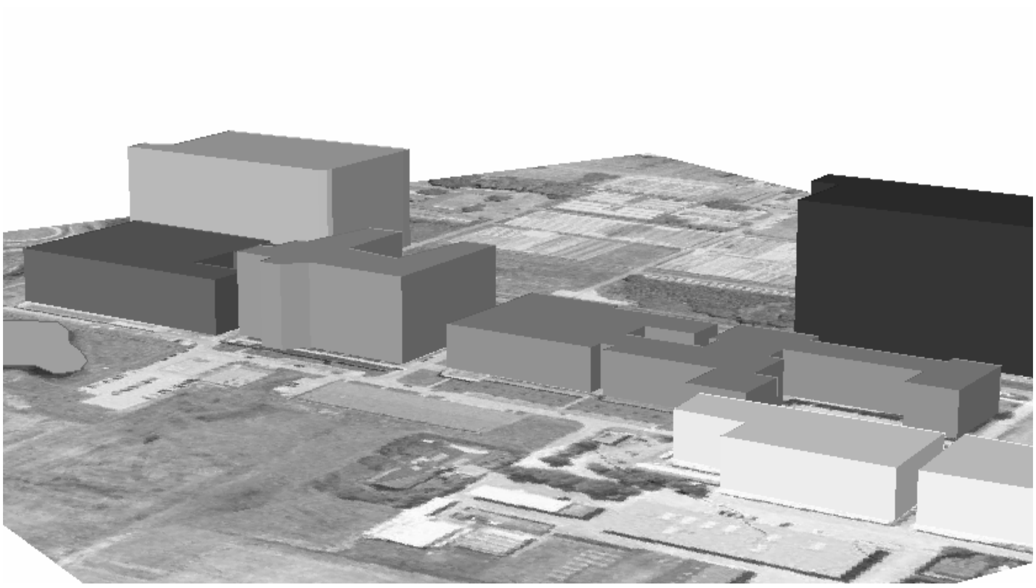


Click on the “Base Heights” tab, if you have a DEM (digital Elevation Model) for this area, click  besides the “Obtain heights for layer from surface:” to open the surface dialog, browse to select the data then add it. Click **Ok** to close the Layer properties dialog.

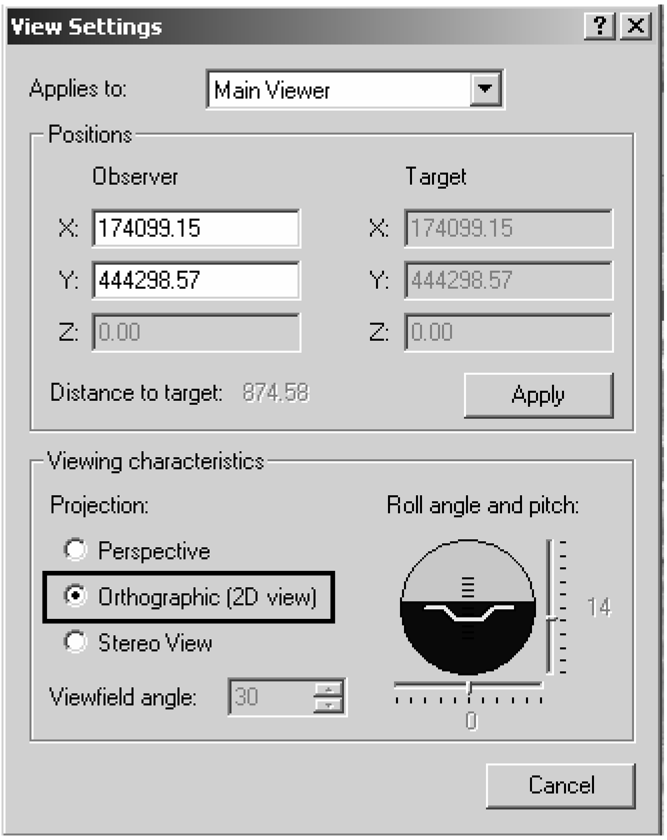




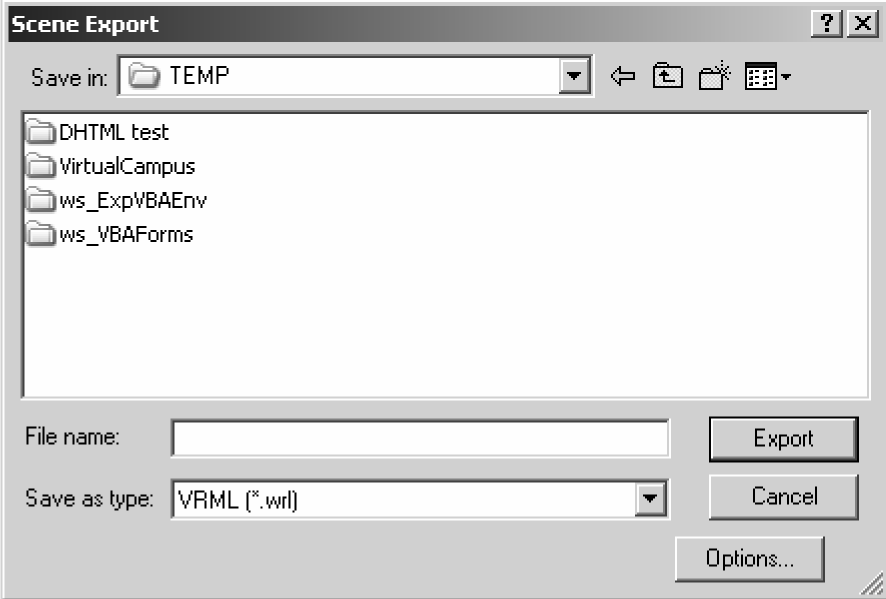
13. Repeat step 13 to extrude more buildings or other objects. Here is the sample result.



14. In ArcScene, click **View -> view settings…**will open the “view settings” dialog, choose “Orthographic (2D view)”, and then close this dialog. You will see the layers in 2D view as you see in ArcMap.



15. Click **File –> Export Scene –> 3D** to open the “Scene Export” dialog, write down the name **3D_features** (don’t change it), save it into: **C:\Temp\visulization\visual_serverfiles.**



16. Edit the 3D_features.wrl by notepad or vrmlpad. Find the "NavigationInfo" and "Viewpoint", delete them. Add "**Inline {url "position.wrl"}"** to their place. Save the file

```
NavigationInfo
{
  headlight FALSE
  speed 28.530451
}

Viewpoint
{
  fieldOfView  0.3
  orientation  1 0 0 -0.785398
  position 0.000000 1164.319336 1141.218018
  description "default"
}
```

```
#VRML V2.0 utf8

WorldInfo
{
  title "ArcScene Document"
  info  "Generated by ArcScene"
}

Background
{
  skyColor 0.100000 0.300000 0.900000
}

DirectionalLight
{
  ambientIntensity 1.0
  direction 0.612372 -0.500000 0.612372
}

Inline {url "position.wrl"}
#****************************************************
```

The step 17 will edit the 3D_features.wrl file in order to make the 3D objects access to the server. Step 18 will edit the server database and ASP file. If the server is not ready, skip these two steps and then **Copy** 3D_features.wrl to**:** **C:\Temp\visualization\3Dvisulization** then Go to step 19.

17. Find the codes which corresponded to the buildings or objects. My way to do it is Toggle comments on some codes and then some objects will disappear. See the results then match them! The codes for the buildings always like this:

```
Group   # (Layer node)
{
 children
 [
  Shape
  {
   appearance
    Appearance
    {
     material
      Material
      {
       ambientIntensity 0.400
       diffuseColor     0.400 0.600 0.804
       emissiveColor    0.000 0.000 0.000
       shininess        1.000
       specularColor    0.000 0.000 0.000
       transparency     0.000
      }
    }  # end appearance
   geometry
    IndexedFaceSet
    {
     ccw   FALSE
     solid FALSE
     coord
      Coordinate
      {
      }
     coordIndex
     [
       0 1 2 -1 3 4 5 -1 6 7 8 -1 9 10 11 -1 12 13 14 -1 15 16 17 -1 18 19
     ]
     creaseAngle 0.0
    } # end face set
  } # end shape (face set)
 ]
}
```

Change to code in the black frame into:

Anchor #(Layer node)

*{ url "javascript: function pop() { window. open('http://server name /serverfolder /NewWest.asp','newpage','width=200,height=80');};**pop();**}"*

description " "

…...

}

You can find some help about the command "Anchor". Change the name of function as you like. Here we use **pop ()**. The server name and server folder you can ask the server administrator. The sample result:

```
Anchor   # (Layer node)   New University building West
{   url "javascript:function pop2() {window.open('http://esgt2391/visualization3D/NewWest.asp','newpage',
    description ""
    children
      [
       Shape
       {
        appearance
         Appearance
         {
          material
           Material
           {
            ambientIntensity 0.400
            diffuseColor     0.663 0.000 0.902
            emissiveColor    0.000 0.000 0.000
            shininess        1.000
            specularColor    0.000 0.000 0.000
            transparency     0.000
           }
         }  # end appearance
        geometry
         IndexedFaceSet
         {
          ccw   FALSE
          solid FALSE
          coord
           Coordinate
           {
           }
          coordIndex
          [
            0 1 2 -1 3 4 5 -1 6 7 8 -1 9 10 11 -1 12 13 14 -1 15 16 17 -1 18 19 20 -1 21 22 23 -1 24 25
          ]
          creaseAngle 0.0
         } # end face set
       } # end shape (face set)
      ]
    }
```

Here esgt2391 is my server name and the files are stored in visualization3D folder. Follow this step until you changed all the codes. Save this file. Compress this file as **3D_feautures.zip** file.

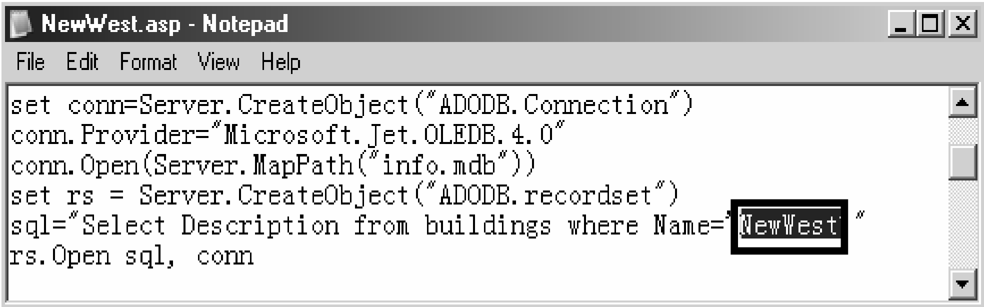
Copy all the files in visulization_server to the proper server folder. Normally, it is the **C:\inetpub\wwwroot\.** Create a new folder name “visulization3D” here, **paste** the files into this folder. Keep this folder open!

18. Open Info.mdb in the folder “vusulization3D”, which is a Microsoft Access file. There is a table called “Buildings”. Double click on it and you will the records.

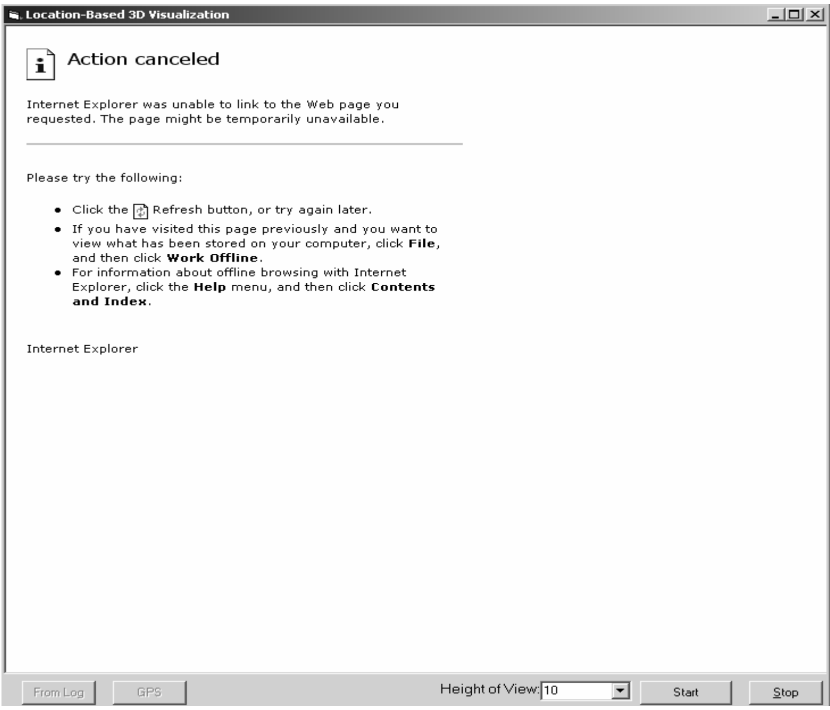
Buildings

ObjID	Name	Description
1	Alterra	Alterra in Wagenigen. New name: GAIA
2	WestAlterra	Enviroment building on the west of Alterra
3	EastAlterra	Library and Canteen. New name: LUMEN
4	NewWest	New Unversity building. It will be the main building from 2006
5	NewEast	New University building face to LUMEN

Edit the name of some buildings and their descriptions. Here we only provide the text information. If you finished editing, you should save your results. You also see the some files with .asp extension in this folder with the file name the same as the name in the table above. If you add some new records in the database, copy one of the .asp file and paste it in the same folder, then rename it with the same name as the name in name field of “buildings” table. Open this new .asp file by Notepad; change the name in the black frame to the file name without .asp extension. Save and close the file.



19. Open folder **C:\temp\visulization\3Dvisualization**, click the **vis3D.exe** to run the application. If you didn’t do step 17, you will see:



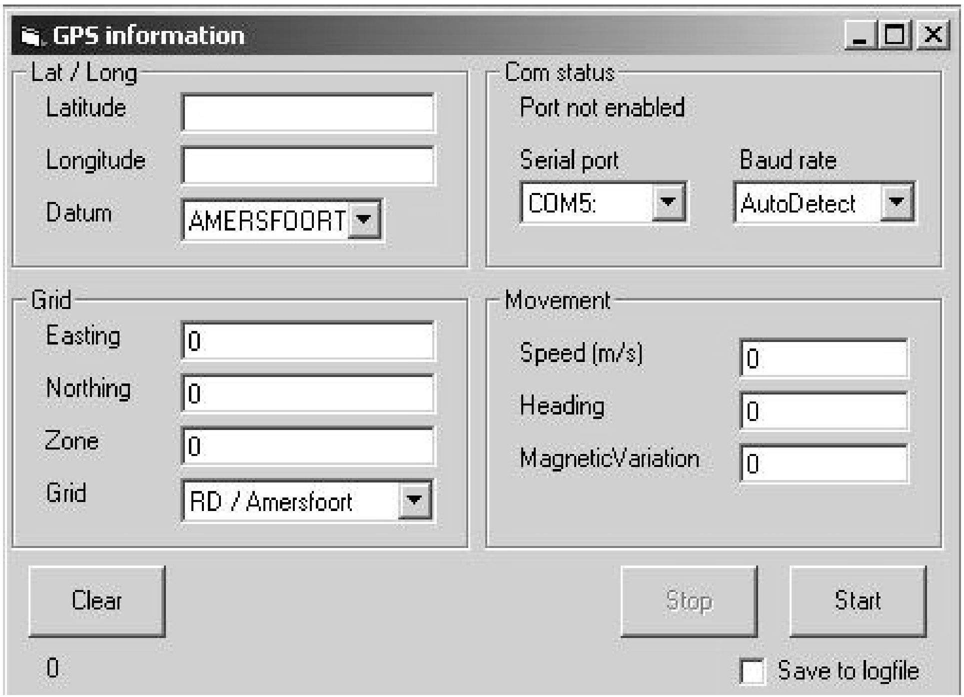
Nothing is displayed in the frame because the information on the server is unreachable. Remember you have copied the 3D_features.wrl to this folder **C:\temp\visulization\3Dvisualization**.

If the server is working, you will see:

Click on the link and download the 3D data and unzip it to the **C:\temp\visualization\3Dvisulization**. Click **Start** button. You will see your 3D model. See the sample model. You can change you eye level by choosing the number beside "Height of View". This interface is the main one.

20. Change your computer system time back to 23/11/2005 because of the license restriction (it's a free open source, but you need ask for license key after one month). Open your GPS device (if you don't have it now or you are in the building, go to step 22), Click the **GPS** button to open the "GPS information" dialog.

21. Choose the right **Datum** and **Grid**. And set the **serial port**. Click **Start** then the application will try to connect to GPS. You can see the status at the upper right of this dialog. **Don't close or minimize** this dialog. Switch to the main one as the picture above. Now you can start to move. See how it works!!!! You can track your route by click the "Save to logfile" checkbox.



22. If you don't have the GPS device now or you are in the building, there is a demonstration to show how it works. Click the **From Log** button (still change the system time) and you will open the "GpsTools log" dialog. Click on the button **Read sentence from log**, you will see how the 3D model display in the main frame changes.